Siemens TCP/IP Ethernet Driver Help

© 2012 Kepware Technologies

Table of Contents

Table of Contents	2
Siemens TCP/IP Ethernet Driver Help	4
Overview	4
Device Setup	5
General Communications Parameters.	5
S7 Communications Parameters	5
Addressing Options	7
How To Configure Connections in Micro/WIN.	8
How To Configure S7-300/400 Connections in STEP 7	5
How to Configure S7-1200 Connections with the Totally Integrated Automation (TIA) Portal	9
Cable Diagrams	5
Optimizing Siemens TCP/IP Ethernet Communications	7
Data Types Description 3	8
Address Descriptions 3	9
S7-200 Address Descriptions	9
S7-300 Address Descriptions	1
S7-400 Address Descriptions	1
S7-1200 Address Descriptions	1
NetLink: S7-300 Address Descriptions	1
NetLink: S7-400 Address Descriptions	2
Internal Tags	2
Standard S7-300/400/1200 Item Syntax	2
Applicom Direct-Link SW1000 Item Syntax4	6
INAT OPC Server TCPIPH1 Item Syntax	8
Siemens Simatic Net Item Syntax	0
Siemens STEP 7 Item Syntax5	2
Softing S7/S5 OPC Server Item Syntax	4
Legacy S7-300/400 Item Syntax	6
Error Descriptions	2
Error Codes	2
Address Validation	3
Missing address	3
Device address ' <address>' contains a syntax error</address>	3
Address <address>' is out of range for the specified device or register</address>	3
Data Type ' <type>' is not valid for device address '<address>'</address></type>	4
Device address ' <address>' is Read Only6</address>	4
Array size is out of range for address ' <address>'</address>	4
Array Support is not available for the specified address: ' <address>'</address>	4
Driver Error Messages	4

١	Winsock initialization failed (OS Error=n)	64
١	Winsock V1.1 or higher must be installed to use the Siemens TCP/IP Ethernet device driver	65
D	Device Status Messages	65
•	'Device <device name="">' is not responding</device>	65
ı	Unable to connect to device ' <device name="">'</device>	65
ı	Unable to establish association with device ' <device name="">'</device>	66
ı	Unable to read <block size=""> bytes starting at address <address> on device '<device name="">'</device></address></block>	67
ı	Unable to write to ' <address>' on device '<device name="">'</device></address>	68
Inde	ex	70

Siemens TCP/IP Ethernet Driver Help

Help version 1.032

CONTENTS

Overview

What is the Siemens TCP/IP Ethernet Driver?

Device Setup

How do I configure a device for use with this driver?

Optimizing Your Siemens TCP/IP Ethernet Communications

How do I get the best performance from the Siemens TCP/IP Ethernet driver?

Data Types Description

What data types does this driver support?

Address Descriptions

How do I address a data location on a Siemens TCP/IP device?

Error Descriptions

What error messages does the Siemens TCP/IP Ethernet driver produce?

Overview

The Siemens TCP/IP Ethernet Driver provides an easy and reliable way to connect Siemens TCP/IP Ethernet devices to OPC Client applications, including HMI, SCADA, Historian, MES, ERP, and countless custom applications. It is intended for use with Siemens S7-200, 300, 400, and 1200 PLCs. There are two options for communications:

- Industrial Ethernet TCP/IP interface communication processor (CP). The protocol used is S7 Messaging on Industrial Ethernet (ISO 8073 Class 0) over TCP/IP as defined in RFC1006.
- Hilscher's NetLink adapter. Only an MPI port is required. The Netlink adapter does not support the S7-200 model.

The driver requires no special libraries or hardware. A standard Ethernet card is all that is needed.

Device Setup

Supported Devices

S7-200 via CP243 S7-300 via CP343 S7-400 via CP443 S7-1200* S7-300 via NetLink S7-400 via NetLink

*The S7-1200 device has a built-in Ethernet module.

Connection Timeout

This parameter specifies the time that the driver will wait for a connection to be made with a device. Depending on network load, the connect time may vary with each connection attempt. The default setting is 3 seconds. The valid range is 1 to 30 seconds.

Request Timeout

This parameter specifies the time that the driver will wait on a response from the device before giving up and going on to the next request. Longer timeouts only affect performance if a device is not responding. The default setting is 2000 milliseconds. The valid range is 100 to 30000 milliseconds.

Retry Attempts

This parameter specifies the number of times that the driver will retry a message before giving up and going on to the next message. The default setting is 2 retries. The valid range is 1 to 10.

Device IDs

Up to 1024 devices may be defined on a given channel. The Device ID is formatted as YYY.YYY.YYY.YYY, where YYY designates the device IP address. Each YYY byte should be in the range of 0 to 255. If the device supports host name resolution, the Device ID may also be specified as a standard UNC/DNS name.

Note: For NetLink users, NetLink communication parameters (such as IP Address, Subnet Mask and Baud Rate) can be configured using the **NetLink Configuration Utility**. This application is located in the server's **Utilities** sub-directory and can be launched using the Start Menu shortcut.

General Communications Parameters

Port Number

This parameter specifies the port number that the remote CP is configured to use.

Default Port

IE TCP/IP: 102 (TSAP) NetLink: 1099

It is recommended that the default port is used for most normal applications where the OPC Server and the PLC exist on the same network. For an application that will be using the Internet through firewalls and advanced routers, the port number can be changed to allow these operations to occur. In most cases, however, the PLC will only accept a connection on port 102/1099 and may require router forwarding.

MPI ID (NetLink Only)

The MPI ID is configured for the port in which the NetLink adapter is connected. This setting does not apply to models utilizing the IE TCP/IP CPs (S7-300, and S7-400). A maximum of 2 connections (or devices) via TCP are possible when using the NetLink adapter.

S7 Communications Parameters

S7-200 Communications Parameters

There are two ways the Siemens TCP/IP Ethernet Driver can communicate to the S7-200 device on an Ethernet network.

- PG Connection (i.e. Connection utilized by Micro/WIN). 1 connection is available.
- Configured Connection (i.e. Connection configured in Micro/WIN via the Ethernet Wizard). 8 connections are available.

Note: Configured Connections are recommended because they free the PG port for Micro/WIN and also provide flexibility to make multiple concurrent connections.

Local TSAP

Link Type	TSAP Value (hex)
PG	4B57 ('KW')
Configured	A Remote (Client) TSAP configured in Micro/WIN's Ethernet Wizard. If Micro/WIN Remote TSAP=xx.yy*, Set Local TSAP to xxyy.

Remote TSAP

Link Type	TSAP Value (hex)
PG	4B57 ('KW')
Configured	A Local (Server) TSAP configured in Micro/WIN's Ethernet Wizard.
	If Micro/WIN Remote TSAP=xx.yy*, Set Local TSAP to xxyy.

*TSAP as displayed in Micro/WIN's Ethernet Wizard. When accessed from V memory, the value may be in decimal form. For example, If TSAP is 10.00; the V memory value will be 1000 hex or 4096 decimal. The values entered for Local TSAP must be in hexadecimal notation; thus, in this example, the value 1000 would be entered.

Rule Of Thumb (from the perspective of the OPC Server)

Local TSAP==Micro/WIN Remote TSAP Remote TSAP==Micro/WIN Local TSAP

For details on using the CP243-1 module and more information in general, refer to **How to Configure S7-200 Connections in Micro/WIN**.

S7-300/S7-400 Communications Parameters

This setting does not apply to models utilizing the NetLink adapter (NetLink: S7-300 and NetLink: S7-400).

Link Settings

The communication link refers to the connection between the Siemens TCP/IP Ethernet Driver and the CP.

Туре

The type of link chosen determines the number of simultaneous requests allowed. The greater the number of simultaneous requests, the greater the data throughput. Each device connection is allowed one outstanding request. To achieve multiple simultaneous requests, multiple connections must be configured. This is achieved by defining the device multiple times in the OPC server (identical device properties). The devices can be defined within the same channel or under separate channels. For more information, refer to **Optimizing Siemens TCP/IP Ethernet Communication**.

Channel.Device=1 CP Connection

There are three types of links: PC (applications), OP (operator panel) and PG (programming device). OP and PG are usually reserved but can be used if all PC connections are taken.

Туре	S7-300 CPU 314, 315	S7-400 CPU 412, 413	S7-400 CPU 414	S7-400 CPU 416
PC	2	14	30	62
OP	1	1	1	1
PG	1	1	1	1

Default Number Simultaneous Requests

Example

Given an S7-400 CPU 412, 14 simultaneous requests can be achieved by defining 14 identical devices in the OPC Server with all configured for Link Type PC. In addition to the PC connections, two more devices can be configured for Link Type OP and PG. This provides 16 connections overall.

Caution: Connection resources are shared amongst applications communicating with the CP. If another application such as STEP 7 is configured to use Industrial Ethernet over TCP/IP, at least one PG/PC connection must be left open for that application to use.

Note: For information on increasing the number of PG, OP and PC type connections, refer to <u>How to Configure</u> S7-300/400 Connections in STEP 7.

CPU Settings

The following settings must match the values entered in STEP 7's HW Configuration program.

Rack

CPU Slot

This parameter specifies the number of the slot in which the CPU of interest resides. To read/write the slot number using an internal tag, refer to **Internal Tags** for more information.

Addressing Options

Addressing Options is used to set the byte order for 16 bit and 32 bit values. Click to select either **Big Endian (S7 Default**) or **Little Endian**.

New Device - Addressing O	ptions	×
	Select the byte order for 16 and 32 bit values. Big Endian (Motorola) is the default byte order for the Siemens S7 controllers. Little Endian (Intel) is available as an option.	
	Byte Order Big Endian (S7 Default) D Little Endian	
	Back Next> Cancel Help	_

Note: Big Endian uses bytes ordered from highest to lowest. Little Endian uses bytes ordered from lowest to highest. With either of these methods the bit order is never changed.

Big Endian

															0)Wo	rd 1														
7	6	5	4	3	2	1	0	15	14	13	12	11	10	9	8	23	22	21	20	19	18	17	16	31	30	29	28	27	26	25	ſ
							N	/ord	1														Wo	rd 3							
7	6	5	4	3	2	1	0	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	15	14	13	12	11	10	9	ſ
	Byte 1 Byte 2									Byte 3 Byte 4																					
7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	Ĺ

Bits

1. The bit range for DWord 1 is 31-0.

2. The bit range for Word 1 and Word 3 is 15-0.

3. The bit range for Byte 1, Byte 2, Byte 3 and Byte 4 is 7-0.

Little Endian

														DW	ord	1														
31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1
			3	3			Wo	rd 3	3			3		3			3	3				W	ord	1		3				
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1
Byte 4 Byte 3								Byte 2 Byte									e 1	e 1												
7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1

Bits

1. The bit range for DWord 1 is 31-0.

2. The bit range for Word 3 and Word 1 is 15-0.

3. The bit range for Byte 4, Byte 3, Byte 2 and Byte 1 is 7-0.

How To Configure Connections in Micro/WIN

Configured Connections are accomplished through the Ethernet Wizard in Micro/WIN. The following instructions illustrate each step in the Ethernet Wizard and also describe any precautions that should be taken as well. Follow these instructions closely in order to use Configured Connections with the Siemens TCP/IP Ethernet Driver correctly.

Note: The Micro/WIN software may require an upgrade before the Ethernet Wizard is made available.

Step 1: Launching the Ethernet Wizard

1. In the Micro/WIN main menu, click Tools | Ethernet Wizard.



2. Then, click Next.



3. Click **Yes** to proceed.



Note: The program must be compiled before the Ethernet Wizard can execute. Correct any errors in the program before continuing.

Step 2: Setting CP243-1 Module Position

1. Click Read Modules.

Ethernet Wizard	×
	This wizard will help you define the parameters for the CP243-1 Ethernet 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 Ethernet modules.
Industrial	Module Position Image: Bead Modules Position Module ID Image: I
	<prev next=""> Cancel</prev>

Note: While it is recommended that the **Read Modules** function be used, this does require that the PLC be connected to the PC either serially or by Ethernet. In either case, the communications parameters for **Micro/WIN** must be properly set for the Read Modules operation to occur.

2. To view the results of Read Modules, select the Ethernet module. Click Next.

Ethernet Wizard	×
Ethernet Wizard	This wizard will help you define the parameters for the CP243-1 Ethernet 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 Ethernet modules. Module Position Module Position Image: CP243-1 Ethernet module Module Position
	<prev next=""> Cancel</prev>

Step 3: Assigning Module Address

1. Enter the **IP Address**, **Subnet Mask** and **Gateway Address** if applicable. Alternatively, enable **BOOTP** if applicable.

2. Select **Auto Detect Communications** in order to allow the module to automatically select either **10BaseT** or **100BaseT**. In rare cases where there may be a cable issue that does not allow the module to operate properly at 100BaseT, force the module to use 10BaseT. This will increase the modules' tolerance to a bad Ethernet line.

Ethernet Wizard	<u>×</u>
	Module Address Please select the address to assign to this CP243-1 module. If your network provides a BOOTP server (a service that will automatically assign IP addresses at startup), you may choose to have an IP address automatically assigned. IP Address: Image: Ima
	<prev next=""> Cancel</prev>
	Auto Detect Communications

3. The image shown below contains demonstration values.

Ethernet Wizard
Image: Second
<prev next=""></prev>

Step 4: Number of Configured Connections

1. Enter the number of desired available connections for this device. If 0 is entered, the only connection available will be the PG connection used by Micro/WIN.

Note: The number of connections selected determines how many simultaneous connections the PLC can support. When intending to have only one OPC Server talking to the PLC, set up only one connection. This will ensure the best performance for the OPC Server. When intending to have more than one active connection to the PLC, use multiple connections. Keep in mind, however, that the performance of the module will be impacted as each connection is used.

2. The image shown below contains 4 connections.

Ethernet Wizard	×
Ethernet Wizard	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. QB 2 =
	<pre></pre>
	<prev next=""> Cancel</prev>

Step 5: Configuring Connections

Each connection is configured individually. For this example, 4 connections have been selected.

Step 5a: Connection 0

There are two types of connections, Client and Server. In a Client Connection, the device is a client and makes request with servers (i.e. other devices). In a Server Connection, the device is a server and handles requests from clients (such as the OPC Server and other devices). The latter is required for communications with the Siemens TCP/IP Ethernet Driver.

1. Select This is a Server Connection....

13

onfigure Connections
You have requested 4 connection(s). For each connection, specify whether the connection should act as a client or server, and configure its associated properties.
Connection 0 (4 connections requested) This is a Client Connection: Client connections request data transfers between the local PLC and a
This is a Server Connection: Servers respond to connection requests from remote clients.
Local Properties (Client) TSAP 10.00 10.00 10.00
You may define up to 32 data transfers between this connection and the remote server. Specify the IP address of the server for this connection. Data Transfers 0 . 0 . 0 . 0 . 0 . 0 . 0 . 0 . 0 . 0 .
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.
Connection0_0
< Prev Connection Next Connection >
OK Cancel

2. From this standpoint, the **CP243-1** is considered the **Server** (Local) and the **OPC Server Channel**. The device is considered the **Client (Remote)**.

3. Enter a **Remote TSAP** or accept the default. This will be the **Local TSAP** in the **OPC Server**.

4. **Optional:** Accept all connection requests or limit to a particular remote machine. It is recommended that **Accept all connection requests** be selected. If concerned about device security (or if intending to access this device over the Internet) select a specific IP address. Users must ensure that the OPC server is running on a PC that has a known and fixed IP address.

5. Select **Enable the Keep Alive**.

6. Click Next Connection.

onfigure Connections	2
You have requested 4 connection(s). For each conne client or server, and configure its associated properties	ction, specify whether the connection should act as a
Connection 0 (4 connections requested)	,
 This is a Client Connection: Client connections remote server. 	quest data transfers between the local PLC and a
 This is a Server Connection: Servers respond to 	connection requests from remote clients.
Local Properties (Server) TSAP 10.00 This server will connection with an Operator Panel (0P)	Remote Properties (Client) TSAP 10.00
Accept all connection requests.	
Accept connection requests from the following clients only:	
<u>عا</u>	
☑ Enable the Keep Alive function for this connection	n.
Please specify a symbolic name for this client connect symbolically when initiating data transfers with the rer	stion. Your program can reference this connection note server.
	< Prev Connection Next Connection >
	OK Cancel

Step 5b: Connection 1

1. Select This is a Server Connection...

Configure Connections	×	
You have requested 4 connection(s). For each connection (s), For each c	ction, specify whether the connection should act as a	
Connection 1 (4 connections requested)		
This is a Client Connection: Client connections request data transfers between the local PLC and a remote server.		
C This is a Server Connection: Servers respond to c	connection requests from remote clients.	
Local Properties (Client) TSAP 11.00	Remote Properties (Server) TSAP 10.00	
You may define up to 32 data transfers between this connection and the remote server. Data Transfers	Specify the IP address of the server for this connection.	
Enable the Keep Alive function for this connection	n.	
Please specify a symbolic name for this client connec symbolically when initiating data transfers with the rem	tion. Your program can reference this connection note server.	
Connection0_1		
	< Prev Connection Next Connection >	
	OK Cancel	

- 2. Notice the **Local TSAP** automatically incremented to 11.00.
- 3. Enter a **Remote TSAP** or accept the default. This will be the **Local TSAP** in the **OPC Server**.
- 4. Optional: Accept all connection requests.
- 5. Select Enable the Keep Alive.
- 6. Click Next Connection.

Configure Connections
You have requested 4 connection(s). For each connection, specify whether the connection should act as a client or server, and configure its associated properties.
Connection 1 (4 connections requested)
O 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 This server will connection with an Operator Panel (OP). Accept all connection requests. Accept connection requests from the following clients only:
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

Step 5c: Connection 2

1. Select This is a Server Connection...

Configure Connections	×	
You have requested 4 connection(s). For each connection(s) for each connection of server, and configure its associated properties.	ction, specify whether the connection should act as a	
Connection 2 (4 connections requested)		
This is a Client Connection: Client connections request data transfers between the local PLC and a remote server.		
C This is a Server Connection: Servers respond to a	connection requests from remote clients.	
Local Properties (Client) TSAP 12.00	Remote Properties (Server) TSAP 10.00	
You may define up to 32 data transfers between this connection and the remote server. Data Transfers	Specify the IP address of the server for this connection.	
Enable the Keep Alive function for this connectio	n.	
Please specify a symbolic name for this client connect symbolically when initiating data transfers with the ren	tion. Your program can reference this connection note server.	
Connection0_2		
	< Prev Connection Next Connection >	
	OK Cancel	

- 2. Notice the **Local TSAP** automatically incremented to 12.00.
- 3. Enter a **Remote TSAP** or accept the default. This will be the **Local TSAP** in the **OPC Server**.
- 4. Optional: Accept all connection requests.
- 5. Select Enable the Keep Alive.
- 6. Click Next Connection.

Configure Connections
You have requested 4 connection(s). For each connection, specify whether the connection should act as a client or server, and configure its associated properties.
Connection 2 (4 connections requested)
O 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 12.00 This server will connection with an Operator Panel (OP). Accept all connection requests. Accept connection requests from the following clients only: E2
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

Step 5d: Connection 3

1. Select This is a Server Connection....

Configure Connections	×	
You have requested 4 connection(s). For each connection (s). For each connection (s), For each connection of server, and configure its associated properties.	ction, specify whether the connection should act as a	
Connection 3 (4 connections requested)		
This is a Client Connection: Client connections request data transfers between the local PLC and a remote server.		
C This is a Server Connection: Servers respond to a	connection requests from remote clients.	
Local Properties (Client) TSAP 13.00	Remote Properties (Server) TSAP 10.00	
You may define up to 32 data transfers between this connection and the remote server. Data Transfers	Specify the IP address of the server for this connection.	
Enable the Keep Alive function for this connectio	n.	
Please specify a symbolic name for this client connect symbolically when initiating data transfers with the ren	tion. Your program can reference this connection note server.	
Connection0_3		
	< Prev Connection Next Connection >	
	OK Cancel	

- 2. Notice the **Local TSAP** automatically incremented to 13.00.
- 3. Enter a **Remote TSAP** or accept the default. This will be the Local TSAP in the OPC Server.
- 4. Optional: Accept all connection requests.
- 5. Select Enable the Keep Alive.
- 6. Click Next Connection.

Configure Connections X
You have requested 4 connection(s). For each connection, specify whether the connection should act as a client or server, and configure its associated properties.
Connection 3 (4 connections requested)
O 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 13.00 This server will connection with an Operator Panel (OP). Accept all connection requests. Accept connection requests from the following clients only:
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

That completes the configuration of the four connections that were selected.

Note: Notice that the **Local TSAP** in the **Connection dialog** was automatically advanced for each connection. This TSAP number will need to be used in the OPC server setup when defining a device as the remote TSAP number.

Step 6: CRC and Keep Alive Interval

Optional: Enable CRC protection to monitor for accidental configuration corruption.

1. Set the **Keep Alive Interval**. The longer the interval, the longer the connection between the device and the OPC Server will exist during idle time. A long Keep Alive Interval may not be desirable if connections are being shared (nonconcurrent). Each remote client will need to wait this amount of time before it will be able to connect with the device once the last connected remote client is finished communications. The 30 second default is suggested.

Ethernet Wizard		×
	 CRC Protection 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. 	
	 Yes, generate CRC protection for this configuration in the data block. No, do not generate CRC protection for this configuration. 	
	 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. <pre> </pre> Next> Cancel	

Step 7: Memory for Configuration

1. Click **Suggest Address** to let the wizard find the best available location to store the **Ethernet configuration**.

Ethernet Wizard		X
	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</prev>	

2. The image shown below displays the results.

Ethernet Wizard		x
	-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 VB159 through VB317	
	<prev next=""> Cancel</prev>	

Note: It is recommended that the Micro/WIN software pick this location for the application. If a CRC was not generated for the configuration data, please take steps to ensure that no other aspect of the PLC program will overwrite this area of memory.

Step 8: Ethernet Wizard Summary

1. Click **Finish** or **Prev** in order to modify the **Configured Connections**.

23



2. To review what the Ethernet Wizard produced, double-click **ETH0_CTRL** under the **Program Block**. All **TSAPs** configured are listed for future reference. Remember, the **Local TSAP** below is the **Remote TSAP** in the OPC Server and the Remote TSAP below is the Local TSAP in the OPC Server.

3. The image below shows ETH0_CTRL.

🖀 STEP 7-N	licro/WIN 32 - tip031b - [SIMAT	IC ST	`L]			
🔣 <u>F</u> ile <u>E</u> dit	_⊻iew_ <u>P</u> LC_ <u>D</u> ebug_ <u>T</u> ools_ <u>W</u> ind	lows	<u>H</u> elp			
<u>n 🖻 🖉 🗍 🗄 🕴 </u>	5 0, X 10 6	V [8 土 工	ÊÎ ₽Î		► ■ 🏧 🖁
View	E Tip031b(CPU 224 REL 01.21)	•••		· · 2 ·		• 3 • • • • • • 4 • • • • • •
	Program Block		Symbol	Var Type	Data Type	Comment
				IN		
Program Blog				IN_OUT		
r rogram bloc	Ten Symbol Table	LU.U	CP_Ready		BUUL	Channel and the
			Ch_Ready		WORD	Error word
4 ,22	🕞 Data Block				WOND	
Symbol Table Status Chart Data Block System Block	System Block Cross Reference Communications Favorites Favorites Favorites Clock Communications Communications Compare Convert Compare Fault Counters Floating-Point Math	•	This POU w module at pr and initialize on every pro command by The followin Connection Connection Connection	as generate osition 0. Th the CP243 ogram scar g connectio 0 is a Serve 1 is a Serve 2 is a Serve 3 is a Serve	ed by the Eth heETHx_CT -1 Ethernet i h, and only b nodule was : ons have be er: Local TS er: Local TS er: Local TS er: Local TS	hernet Wizard for use with a CP243-1 'RL (Control) instruction is used to enable module. This instruction should be called the used once in your program. The specified as QB2. en configured for this module: SAP: 10.00 Remote TSAP: 10.00 Client Addresses: 0.0.00 SAP: 11.00 Remote TSAP: 10.00 Client Addresses: 0.0.00 SAP: 12.00 Remote TSAP: 10.00 Client Addresses: 0.0.00 SAP: 13.00 Remote TSAP: 10.00 Client Addresses: 0.0.00

4. The image below shows ETH0_CFG.



5. Now that the results of the Ethernet Wizard have been confirmed, a connection can be made using the OPC Server.

How To Configure S7-300/400 Connections in STEP 7

In order to configure the S7-300/400 for communications with the Siemens TCP/IP Ethernet Driver, both the CPU and the Ethernet module will need to be configured as well. To do so, follow the directions below.

1. From the **Simatic Manager**, launch **HW Config** by double-clicking **Hardware** under the **SIMATIC Station**.

25



2. If this is a new Simatic project, add the necessary modules to the **Rack** in HW Config. For the Siemens TCP/IP Ethernet Driver to communicate with the CPU, there will need to be at least one Ethernet module capable of **S7 Communications**. This may be built into the CPU.

Configuring an Ethernet Module

1. Right-click on the particular module in the rack and then select **Object Properties**.



2. The dialog should appear as shown below.

Properties - CP 343-1 - ((R0/54)	X
General Addresses Op	ptions Diagnostics	
Short Description:	CP 343-1	
	S7 CP for Industrial Ethernet ISO and TCP/IP with SEND/RECEIVE and FETCH/WRITE interface, long data, UDP, TCP, ISO, S7 communication, routing, module replacement without PG, 10/100 Mbps, fixed MAC address, initialization over LAN, IP multicast, firmware V2.0	<u>←</u>
Order No./ firmware	6GK7 343-1EX11-0XE0 / V2.0	
Name:	CP 343-1	
Interface Type: Ethe	ernet MPI Address: 3	
Address: 10.1	0.110.70	11
Networked: Yes	Properties	
Comment:		
		-
ОК	Cancel Help	

3. From the **General** tab, click the **Interface** | **Properties** button.

Properties - Ethernet interface CP 343-1 (R	:0/54)	×
General Parameters		
E Set MAC address / use ISO protocol		
MAC address:	If a subnet is selected, the next available addresses are suggested.	
IP protocol is being used		
IP address: 192.168.0.1	Gateway	
Subnet mask: 255,255,255,0		
	Address: 192169.0.1	
Subnet	Address. 1132.100.0.1	
not networked	New	
Ethernet(1)		
	Properties	
	Delete	
OK	Cancel Help	

www.kepware.com

- 4. Specify the **IP** and **Subnet Mask** for this module.
- 5. To network this module, click **New** under **Subnet**. Next, select the network created and the click **OK**.
- 6. Return to the HW Config main window.

Configuring Connections

1. Right-click on the CPU module in the rack and select **Object Properties**.

🖳 HW Config - [SIMATIC 3	Disconnect Master System		
🛄 Station Edit Insert PL	Insert PROFINET IO System		
🗅 🗃 🔓 🖳 🚭	PROFINET IO Manage Sync Domain.		
 , ⊊_(0) UB	PROFINET IO Topology		·····································
1 PS 307 .	Isochrone Mode		mt mi
2 CPU 31	Specify Module		
$\frac{\chi_2}{3}$ D^p	Delete	Del	Standard
4 H CP 343- 5	Go To Filter Assigned Modules	•	Rofibus dp Rofibus-pa Rofinet Io
	Monitor/Modify		IMATIC 300
•	Edit Symbols		IMATIC 400 IMATIC PC Based Control 300/400
	Object Properties	Alt+Return	IMATIC PC Station
	Product Support Information	Ctrl+F2	
Slot 🚺 Module	FAQs	Ctrl+F7	
1 PS 307 2A	Find Manual	Ctrl+F6]
X2 DP	DES7Y1.2 Z 1025		
3			
4 🗧 CP 343-1	6GK7 V2.0 3 256 256		
$\left \frac{5}{c} \right $			
9			
10		PROFIB	US-DP slaves for SIMATIC S7, M7, and 🔁 🛓
Displays properties of the selecte	ed object for editing.		

2. The dialog should appear as shown below.

Properties - CPU 315-2 DP - (R0/52)		×
General Startup Cycle/Clock	Memory Retentive Memory	Interrupts
Time-of-Day Interrupts Cyclic Interrupts	Diagnostics/Clock Protection	
Connection Resources Reserved for		
PG Communication:	1	
OP Communication:	1	
S7 Standard Communication:	4	
S7 Communication:	0 (already configured)	
Maximum Number of Connection Resources:	12	
OK	Cancel	Help

3. Configure the desired number of PG/OP and PC (S7 Communication) connections.

Туре	Description
PG Communication	Used for program loading, diagnostics
OP Communication	Used for operator control and monitoring
S7 Standard	Communication connections not configured, MPI communications with PUT/GET function
Communication	blocks
S7 Communication (PC)	Configured connections, data communications

Note: The maximum number of PC connections for the CPU equals the Maximum Number of Connection Resources **minus** the S7 Standard Communication resources **minus** the OP Communication resources **minus** the PG Communication resources. Note that the Maximum Number of Connection Resources is based on the CPU/version/firmware.

In the example shown above, there are six S7 Communication (PC) connections available (12-4-1-1=6). Likewise, the number of PG and OP connections can be increased using the same concept.

If the **Device returned protocol [Class=0x83, Code=0x04]** error is encountered, increase the number of S7 Standard Communication connections, thereby decreasing the number of S7 Communication connections.

4. After the connections have been configured, click **OK**. Next, in the main HW Config window click **Station** | **Save and Compile**.

5. Click **PLC** | **Download** to commit to the changes.

How to Configure S7-1200 Connections with the Totally Integrated Automation (TIA) Portal

In order to configure the S7-1200 for communications with the Siemens TCP/IP Ethernet Driver, an online connection is required between the programming device and the target system. Users may have to configure the programming device to talk to the target system. For more information, follow the instructions below.

Note: For new Simatic projects, refer to the PLC's documentation for information on the default IP address settings.

1. Start the TIA Portal. In the Portal View, click **Create new project**.

Siemens					_ # X
				Tot	tally Integrated Automation PORTAL
Start	I			Create new p	project
Devices &	. 🖈	o 0)pen existing project	Project name:	Project1
Networks				Path:	C:\My Documents\Automation an
DV C		<u> </u>	reate new project	Author:	
Programming		💮 М	ligrate project 🛛 🚽	Comment	î.
Visualization					Treate
Online & Diagnostics	1	O F			
) In	nstalled products		
		💮 н	lelp		
Project view		0)pened project:		

2. Next, select the **Online & Diagnostics** tab and then click **Accessible Devices**.

- 3. Select the appropriate PG/PC interface. This will prompt the TIA to scan the network for the device.

- 4. Once the scan is complete, select the device and then click **Show**. This will invoke the Project View.
- 5. In the project tree, locate the IP address and then open **Online & Diagnostics**.
- 6. Next, double-click **Online & Diagnostics** to invoke **Online Access**.

7. Select **Functions** and then click **Assign IP Address**.



8. Enter the communication settings and click **Assign IP Address**.

Note: The device is now ready to be configured.



9. Return to the Portal View and then select the Device & Networks tab. Then, click Add new device.

10. Next, select the device's configuration and then click **Add device**. This will invoke the Project View, where the device's hardware can be further configured.



11. Once finished, view the project tree. Locate **Program Blocks** and **PLC Tags** and then configure the addresses that will be used in the PLC project.

naject freie 🛛 🔣 🖣	Propert1 + PLC_1 + PLC tags		_##×	Tasks	
Devices			PLC tags Constants	▼Find and Replace	
100	and the BX			End	1
	PLC tags				
Project1	Name Data tipe	Address Retain Corres	ent.	-	
add new device	1.40	2 001100		Obs	
Eences & Hetuonis					
• _ PLC_1[OPU 1212C ADDCRI/]	1.1			Whate words prov	
Crime 5 diamontics				- Match case	
- Congreen blocks				Find in substructures	
Add new block				Final in history same	
Main [OE1]				All and a state of the second	
 Technological Objects 					
• PLC tags				the regular expressions.	
Single and tables				Whale document	
In Test lists				· From current position	
🖡 🧱 Local modules 🛛 🖉	100 million (100 m		-	Calepton	
• af Cemenon data		g Properties	🔁 Info 🔛 Diagnostics 💌	• Down	
Canguages & Resources	Device information				
Containe access	1 Devices with problems			up	
and and card resource	🖞 Online 🤷 Opera - Deviceimodule	Message Details	Help	Fina Fina new	
	at and a full the	not available Establish new online	e connection 7	▼ Languages & Resources	
				Editing language	
					-
					-
				1222 (C -	

Note: The device is now configured and can be placed in Run Mode for communications.

Cable Diagrams

35



BLU/WHT 5

BRN/WHT 7

BRN 8

<u>OR</u>6 RD -

RJ45

Patch Cable (Straight Through)

5 BLU/WHT

7 BRN/WHT

8 BRN

RD-6GRN

RJ45
Optimizing Siemens TCP/IP Ethernet Communications

The Siemens TCP/IP Ethernet driver has been designed to provide the best performance with the least amount of impact on the system's overall performance. While the Siemens TCP/IP Ethernet driver is fast, there are a couple of guidelines that can be used in order to control and optimize the application and gain maximum performance.

Our server refers to communications protocols like Siemens TCP/IP Ethernet as a channel. Each channel defined in the application represents a separate path of execution in the server. Once a channel has been defined, a series of devices must then be defined under that channel. Each of these devices represents a single Siemens TCP/IP Ethernet controller from which data will be collected. While this approach to defining the application will provide a high level of performance, it won't take full advantage of the Siemens TCP/IP Ethernet driver or the network. An example of how the application may appear when configured using a single channel is shown below.



Each device appears under a single Siemens TCP/IP Ethernet channel. In this configuration, the driver must move from one device to the next as quickly as possible in order to gather information at an effective rate. As more devices are added or more information is requested from a single device, the overall update rate begins to suffer.

If the Siemens TCP/IP Ethernet driver could only define one single channel, then the example shown above would be the only option available; however, the Siemens TCP/IP Ethernet driver can define up to 16 channels. Using multiple channels distributes the data collection workload by simultaneously issuing multiple requests to the network. An example of how the same application may appear when configured using multiple channels to improve performance is shown below.



Each device has now been defined under its own channel. In this new configuration, a single path of execution is dedicated to the task of gathering data from each device. If the application has X or fewer devices, it can be optimized exactly how it is shown here. The performance will improve even if the application has more than 16 devices. While 16 or fewer devices may be ideal, the application will still benefit from additional channels. Although by spreading the device load across all 16 channels will cause the server to move from device to device again, it can now do so with far less devices to process on a single channel.

Important: Although the OPC Server limits the number of channels to 16, the device ultimately determines the number of allowed connections. This constraint comes from the fact that some devices cannot support 16 connections. For these devices, the maximum number of channels defined should equal the maximum number of connections allowed. For devices that support more than 16 connections, the maximum 16 channels should be defined, with devices spread evenly over these 16 channels. For more information on device connections, refer to **Link Settings**.

Data Types Description

Data Type	Description
Boolean	Single bit
Byte	Unsigned 8 bit value
Char	Signed 8 bit value
Word	Unsigned 16 bit value
	bit 0 is the low bit bit 15 is the bid bit
Short	Signod 16 bit value
SHUL	Signed to bit value
	bit 0 is the low bit
	bit 14 is the high bit
	bit 15 is the sign bit
BCD	Two byte packed BCD
	Value range is 0-9999. Behavior is undefined for values beyond this
	range
DWord	Unsigned 32 bit value
	hit 0 is the low hit
	bit 31 is the high bit
long	Signed 32 hit value
Long	
	bit 0 is the low bit
	bit 30 is the high bit
	bit 31 is the sign bit
LBCD	Four byte packed BCD
	Value range is 0-999999999. Behavior is undefined for values beyond
Floot	unis range
Float	32 bit floating point value
	The driver interprets two consecutive registers as a floating-point
	value by making the second register the high word and the first
	register the low word.
String	Null terminated ASCII string*

*The Data Block subtype, String, is a NULL padded ASCII string.

Address Descriptions

Address specifications vary depending on the model in use. Select a link from the following list to obtain specific address information for the model of interest.

S7-200 Address Descriptions S7-300 Address Descriptions S7-400 Address Descriptions S7-1200 Address Descriptions NetLink: S7-300 Address Descriptions NetLink: S7-400 Address Descriptions Internal Tags

S7-200 Address Descriptions

The default data types for dynamically defined tags are shown in **bold**.

Address Type	Range	Туре	Access
Discrete Inputs (IEC)	I0.b-I65535.b*	Boolean	Read/Write
	.b is Bit Number 0-7		
	IB0-IB65535	Byte , Char, String**	Read/Write
	1W0-1W65534	Word, Short, BCD	Read/Write
	ID0-ID65532	DWord, Long, LBCD, Float	Read/Write
Discrete Inputs (SIMATIC)	E0.b-E65535.b*	Boolean	Read/Write
	.b is Bit Number 0-7		
	EB0-EB65535**	Byte, Char, String**	Read/Write
	EW0-EW65534	Word, Short, BCD	Read/Write
	ED0-ED65532	DWord, Long, LBCD, Float	Read/Write
Note: I and E access the same memory area.			
Discrete Outputs (IEC)	Q0.b-Q65535.b*	Boolean	Read/Write
	.b is Bit Number 0-7		
	QB0-QB65535	Byte , Char, String**	Read/Write
	QW0-QW65534	Word, Short, BCD	Read/Write
	000-0065532	DWord Long LBCD Float	Read/Write
Discrete Outputs (SIMATIC)	A0 h-465535 h*	Boolean	Read/Write
	.b is Bit Number 0-7	boolean	Reddy Write
	AB0-AB65535	Byte, Char, String**	Read/Write
	AW0-AW65534	Word, Short, BCD	Read/Write
	AD0-AD65532	DWord, Long, LBCD, Float	Read/Write
Note: Q and A access the same memory area.	-		-
Analog Inputs (IEC)	AIO-AI65534***	Word, Short	Read Only
	AIWO-AIW65534		
Analog Inputs (SIMATIC)	AEU-AE65534*** AEW0-AEW65534	word, Snort	Read Only
Note: AI and AE access the same memory area.			
Analog Outputs (IEC)	AQ0-AQ65534*** AQW0-AQW65534	Word, Short	Read/Write
Analog Outputs (SIMATIC)	AA0-AA65534***	Word, Short	Read/Write
	AAW0-AAW65534		
Note: AQ and AA access the same memory area.			
Internal Memory	MU.D-M65535.D*	Boolean	Read/Write

	.b is Bit Number 0-7		
	MB0-MB65535	Byte , Char, String**	Read/Write
	MW0-MW65534	Word, Short, BCD	Read/Write
	MD0-MD65532	DWord, Long, LBCD, Float	Read/Write
Special Memory	SM0.b-SM65535.b*	Boolean	Read/Write
(Bytes 0-29 are Read Only)	.b is Bit Number 0-7		
	SMB0-SMB65535	Byte, Char, String**	Read/Write
	SMW0-SMW65534	Word, Short, BCD	Read/Write
	SMD0-SMD65532	DWord, Long, LBCD, Float	Read/Write
Sequence Control Relay (SCR)	S0.b-S65535.b*	Boolean	Read/Write
	.b is Bit Number 0-7		
	SB0-SB65535	Byte, Char, String**	Read/Write
	SW0-SW65534	Word, Short, BCD	Read/Write
	SD0-SD65532	DWord, Long, LBCD, Float	Read/Write
Variable Memory	V0.b-V65535.b*	Boolean	Read/Write
	.b is Bit Number 0-7		
	VB0-VB65535	Byte, Char, String**	Read/Write
	VW0-VW65535	Word, Short, BCD	Read/Write
	VD0-VD65535	DWord, Long, LBCD, Float	Read/Write
Timer Current Values	T0-T65535*	DWord, Long	Read/Write
Timer Status Bit	T0-T65535*	Boolean	Read Only
Counter Current Values (IEC)	C0-C65535*	Word, Short	Read/Write
Counter Status Bit (IEC)	C0-C65535*	Boolean	Read Only
Counter Current Values (SIMATIC)	Z0-Z65535*	Word, Short	Read/Write
Counter Status Bit (SIMATIC)	Z0-Z65535*	Boolean	Read Only
Note: C and Z access the same memory area.			°
High-Speed Counter	HC0-HC65535*	DWord, Long	Read Only

*These memory types/subtypes do not support arrays

**Byte memory types (i.e. MB) support Strings. The syntax for strings is <address>.<length> where 0 < length <= 212.

***For Analog Inputs and Outputs the address must be even (AI0, AI2, AI4...).

Note 1: All offsets for memory types I, Q, M, S and SM represent a byte starting location within the specified memory type.

Note 2: Use caution when modifying Word, Short, DWord, and Long types. For I, Q, and F each address starts at a byte offset within the device. Therefore, Words MW0 and MW1 overlap at byte 1. Writing to MW0 will also modify the value held in MW1. Similarly, DWord, and Long types can also overlap. It is recommended that these memory types be used so that overlapping does not occur. As an example, with DWord MD0, MD4, MD8 ... and so on can be used to prevent overlapping bytes.

Arrays

All memory types/subtypes with the exception of those marked with an asterisk (*), support arrays. Below are valid syntax for declaring an array. If no rows are specified, row count of 1 is assumed.

<address>[rows][cols] <address>.rows.cols <address>,rows,cols <address>_rows_cols

For Word, Short, and BCD arrays, the base address + (rows * cols * 2) cannot exceed 65536. Keep in mind that the elements of the array are words, located on a word boundary. For example, IW0[4] would return IW0, IW2, IW4, and IW6.

For Float, DWord, Long and Long BCD arrays, the base address + (rows * cols * 4) cannot exceed 65536. Keep in mind that the elements of the array are DWord, located on a DWord boundary. For example, ID0[4] will return ID0, ID4, ID8, ID12.

For all arrays, the total number of bytes being requested cannot exceed the internal block size of 218 bytes.

S7-300 Address Descriptions

Standard Support <u>S7-300/400/1200 Item Syntax</u> Internal Tags

Third-Party Support

For users familiar with the following applications, limited addressing support is available.

Applicom Direct-Link SW1000 Item Syntax INAT OPC-Server TCPIPH1 Item Syntax Siemens Simatic Net Item Syntax Siemens STEP 7 Item Syntax Softing S7/S5 OPC Server Item Syntax

Legacy Support Legacy S7-300/400 Item Syntax

Note: All brand and product names are trademarks, registered trademarks or service marks of their respective holders.

S7-400 Address Descriptions

Standard Support <u>S7-300/400/1200 Item Syntax</u> <u>Internal Tags</u>

Third-Party Support For users familiar with the following applications, limited addressing support is available.

Applicom Direct-Link SW1000 Item Syntax INAT OPC-Server TCPIPH1 Item Syntax Siemens Simatic Net Item Syntax Siemens STEP 7 Item Syntax Softing S7/S5 OPC Server Item Syntax

Legacy Support Legacy S7-300/400 Item Syntax

Note: All brand and product names are trademarks, registered trademarks or service marks of their respective holders.

S7-1200 Address Descriptions

Standard Support <u>S7-300/400/1200 Item Syntax</u> Internal Tags

Third-Party Support

For users familiar with the following applications, limited addressing support is available.

Applicom Direct-Link SW1000 Item Syntax INAT OPC-Server TCPIPH1 Item Syntax Siemens Simatic Net Item Syntax Siemens STEP 7 Item Syntax Softing S7/S5 OPC Server Item Syntax

Legacy Support Legacy S7-300/400 Item Syntax

Note: All brand and product names are trademarks, registered trademarks or service marks of their respective holders.

NetLink: S7-300 Address Descriptions

Standard Support S7-300/400/1200 Item Syntax

Third-Party Support

For users familiar with the following applications, limited addressing support is available.

Applicom Direct-Link SW1000 Item Syntax INAT OPC-Server TCPIPH1 Item Syntax Siemens Simatic Net Item Syntax Siemens STEP 7 Item Syntax Softing S7/S5 OPC Server Item Syntax

Legacy Support Legacy S7-300/400 Item Syntax

Note: All brand and product names are trademarks, registered trademarks or service marks of their respective holders

NetLink: S7-400 Address Descriptions

Standard Support S7-300/400/1200 Item Syntax

Third-Party Support

For users familiar with the following applications, limited addressing support is available.

Applicom Direct-Link SW1000 Item Syntax INAT OPC-Server TCPIPH1 Item Syntax Siemens Simatic Net Item Syntax Siemens STEP 7 Item Syntax Softing S7/S5 OPC Server Item Syntax

Legacy Support Legacy S7-300/400 Item Syntax

Note: All brand and product names are trademarks, registered trademarks or service marks of their respective holders

Internal Tags

Although the following internal tags are not visible in the server configuration, they can be browsed by the OPC client. They can be found under the **<Channel Name>.<Device Name>._InternalTags** group. If the OPC client does not support browsing, or if a non-OPC client is being used, the tags can be created both dynamically and statically by using the addresses given below.

The tags listed in the following table are valid only for the S7-300 and S7-400 device models. The default data types are listed in **bold**.

Device Address	Description	Range	Data Type	Access
_RACK	Number of the rack in which the CPU of interest resides.	0-7	Byte , Short	Read/Write
	On changing this device property, the connection with the CPU is re-established.			
_SLOT	Number of the slot in which the CPU of interest resides.	2-31	Byte , Short	Read/Write
	On changing this device property, the connection with the CPU is re-established.			

Standard S7-300/400/1200 Item Syntax

Address Syntax

Input, Output, Peripheral, Flag Memory Types

```
<memory type><S7 data type><address>
```

```
<memory type><S7 data type><address><.bit>
```

```
<memory type><S7 data type><address><.string length>*
```

```
<memory type><S7 data type><address><[row][>col]>
```

Timer and Counter Memory Types

<memory type><address>

DB Memory Type

```
DB<num>,<S7 data type><address>
DB<num>,<S7 data type><address><.bit>
DB<num>,<S7 data type><address><.string length>*
DB<num>,<S7 data type><address><[row][col]>
```

where <num> ranges from 1 to 65535.

*Applies to S7 Data Types that support String. String length can vary from 0 < n <= 212 with the exception of S7 Data Type String which can vary from 0 < n <= 210.

See Also: Examples and String Support.

Memory Types

Memory Type	Description	Address Range	Data Type	Access
I E	Inputs			Read/Write
Q A	Outputs		Read/Write	
PI PE	Peripheral Inputs	Depends on the S7 Data	Read Only	
PQ PA	Peripheral Outputs	information, refer to the	Read/Write	
M F	Flag Memory	-	Read/Write	
DB	Data Blocks			Read/Write
Т	Timers	T0-T65535	Read/Write	
C Z	Counters	C0-C65535 Z0-Z65535	Word, Short	Read/Write

See Also: Examples

S7 Data Types

The S7 Data Type is used to coerce the data type for a tag. It does not apply to Timers and Counters. The default data types are shown in **bold**.

S7 Data Type	Description	Address Range	Data Type
Х	Bit	X0.b-X65534.b .b is Bit Number 0-15	Boolean
B Byte	Unsigned Byte	B0-B65535 BYTE0-BYTE65535	Byte , Char
		B0.b-B65535.b BYTE0.b-BYTE65535.b .b is Bit Number 0-7	Boolean
		B0.n-B65535.n BYTE0.n-BYTE65535.n .n is string length. 0 < n <= 212.	String*
C Char	Signed Byte	C0-C65535 CHAR0-CHAR65535	Byte, Char
		C0.b-C65535.b CHAR0.b-CHAR65535.b .b is Bit Number 0-7	Boolean
		C0.n-C65535.n CHAR0.n-CHAR65535.n .n is string length. 0 < n < = 212.	String*
W	Unsigned Word	W0-W65534	Word, Short, BCD

Word W0RD0-W0RD65534 Boolean I Signed Word IO-165534 Boolean INT Signed Word IO-165534 Word, Short, BCD INT Double Word IO-165534 Boolean D Unsigned D0-55532 Boolean DWORD Dusigned D0-565532 Boolean DWORD Double Word D0-565532 DWord, Long, LBCD, Float DI Signed D0-b-055322 DWord, Long, LBCD, Float DIT Signed D10-D165532 DWord, Long, LBCD, Float DIT Signed D10-D165532 DWord, Long, LBCD, Float DIT Signed D10-D165532 DWord, Long, LBCD, Float DIT Sting Double Word D10-D165532 Boolean DIT S7 Date And Time. D10-D165528 String DT S7 Date And Time. DT0-D1765528 String Displayed as string format "m/d/y h:mm:ss cAM/PM> "with range "11/1990 0:0:00 AM" to "12/31/2089 23:59:59 PM". REAL0-REAL65532 Float REAL IEEEF Toot REAL0-REAL65532 Float String S7 String STRING0.n-STRING65532 n. is string length. ocn <= 210. String TOD S7 Time_Of_Day. TOD0-TOD65532 String <th></th> <th>1</th> <th></th> <th></th>		1			
Image: signed word W0.b-W65534.b W0RD0.b-W0RD65334.b .bis Bit Number 0-15 Boolean INT Signed Word ID-165534 INT0-INT65534.b .bis Bit Number 0-15 Word, Short, BCD D Db-165534.b .bis Bit Number 0-15 Boolean D Do-165532.b .bis Bit Number 0-15 Boolean DWORD Unsigned Double Word D0-165532.b .W0RD0-DW0RD65532.b .bis Bit Number 0-31 DWord, Long, LBCD, Float DI DINT Signed Double Word D0-065532.b .D0.b-D65532.b .D0.b-D165532.b .D000000.b-W0RD65532.b .bis Bit Number 0-31 Boolean DI DINT Signed Double Word D10-D165532.b .D100-D10765532.b .D1000000.b .D1070.0-D10765532.b .D100000000.0 D1070.0-D10765532.b .D100000000.0 D1070.0-D10765532.b .D1000000000.0 D1070.0-D10765532.b .D1000000000.0 D1070.0-D10765532.b .D1000000000.0 D1070.0-D10000000000000000000000000000000000	Word		WORD0-WORD65534		
I WUD.0-W05534.b .b is Bit Number 0-15 Boolean INT Signed Word IO-165534 INTO-INT65534.b INTO-INT65534.b INTO-DWORD65532.b .b is Bit Number 0-15 Word, Short, BCD D Unsigned Double Word D0-565532 DWORD0.b-DWORD65532.b .b is Bit Number 0-31 DWord, Long, LBCD, Float DI DWORD Signed Double Word D10-D65532 DWORD0.b-DWORD65532.b .b is Bit Number 0-31 DWord, Long, LBCD, Float DI DI DI DI DI DI DT Signed Double Word D10-D165532.b DUNT0.b-D1NT65532.b .b is Bit Number 0-31 DWord, Long, LBCD, Float DT S7 Date And Time. D10-D165532.b DINT0.b-D1NT65532.b .b is Bit Number 0-31 Boolean DT S7 Date And Time. D10-D165532.b DINT0.b-D1NT65532.b .b is Bit Number 0-31 String DT S7 Date And Time. D10-D165528 String Complex data type stored with 8 bytes as follows: O year, 1 month, 2 days, 3 hours, 4 minutes, 5 seconds, 6 tow most significant digits of MSEC, 7 (4MSB) two ileast significant digits of MSEC, 7 (4MSB) two ileast si					
I Norodo Dr. WORDO 3534.0 Lis Bit Number 0-15 bis Bit Number 0-15 INT Signed Word ID-165534 INT ID-165534.5 Boolean D Unsigned D0-D65532 DWORD Double Word D0-D65532. DWORD Duble Word D0-D65532. DI Double Word D10-D165532. DI Digned D10-D165532. DIT Signed D10-D165532. DINT D10-D165532. Boolean DIT Signed Double Word D10-D165532. DINT D10-D165532. Boolean DIT S7 Date_And_Time. D10-D165532. Complex data type stored with 8 bytes a follows: 0 year, 1 month, 2 days, 3 hours, 4 minutes, 5 seconds, 6 two most significant digits of MSEC, 7 (4MSB) two least significant digits of MSEC				Boolean	
I Jb is Bit NUMBER 0-15 Word, Short, BCD INT Signed Word ID-165534 INTO-INT65534.b Boolean D Unsigned D0-055532 DWORD, DWORD65532.b Boolean D Double Word D0-055532.b DWORD, DWORD65532.b Boolean DI Double Word D10-0155532.b DWORD, b-DWORD65532.b Boolean DI Signed D10-0155532.b DWOrd, Long, LBCD, Float DI Double Word D10-0155532.b DWOrd, Long, LBCD, Float DINT Signed D10-0155532.b DINT0-DINT65532.b DINT Double Word D10-D1055532.b Boolean DIT S7 Date_And_Time. DT0-DT65528 String Complex data type stored with 8 bytes as follows: 0 year, 1 month, 2 days, 3 hours, 4 minutes, 5 seconds, 6 two most significant digits of MSEC, 7 (4MSB) two least significant digits of MSEC, 7 (4LSB) day of week (1=Sunday). String Displayed as string format "m/dy/ h:mm:ss AM/PMS" with range "1/2/31/2089 23:59:59 PM". ReALO-REAL65532 Float REAL IEEE Float REALO-REAL65532 String Stored as DWORD, representing milliseconds since midnight. Displayed as String format "h:m:s.mmm" with range "0:0:0:0:0" to "23:59:59.999". String TOD S7 Time_Of_Day. TOD0-TOD65532 String			WORDU.D-WORD65534.D		
INT Signed Word I0-65534 Word, Short, BCD INT INTO. INTG5534 Boolean D Unsigned D0-065532. DWord, Long, LBCD, Float DWORD Double Word D0-065532. DWord, Long, LBCD, Float DI Signed D00-0000065532. DWord, Long, LBCD, Float DI Signed D00-015532. DWord, Long, LBCD, Float DI Signed D00-015532. DWord, Long, LBCD, Float DINT Signed D10-0165532. DWord, Long, LBCD, Float DINT Softend D10-0165532. DWord, Long, LBCD, Float DI String S7 Date_And_Time. D10-0165532. Boolean DT S7 Date_And_Time. D10-0165532. Boolean O year, 1 month, 2 days, 3 hours, 4 minutes, 5 seconds, 6 two most significant digits of MSEC, 7 (4MSB) two least significant digits of MSEC, 7 (4MSB) two least significant digits of MSEC, 7 (4MSB) two interst significant digits of MSEC, 7 (4MSB)			.D IS BIT NUMBER 0-15		
INT INTO-INTES534 Boolean DWORD Unsigned Double Word D0-065532. DWORD0-DWORD65532. DWORD0.b-DWORD65532. DWORD0.b-DWORD65532. DWORD0.b-DWORD65532. DUNTO-DIVT65532. DINT DWord, Long, LBCD, Float DI DI DI DI DI DI DI DI DI DT Signed Double Word D10-D165532 DIVTO-DIVT65532. DIVTOD DIVTO-DIVT65532. DIVTOD DIVTODITF0. DIVTODITF0. DIVTODITF0. DIVTODITF0. DIVTODITF0. DIVTODI	I	Signed Word	10-165534	Word, Short , BCD	
ID.b-165534.b INTO.b-INT65534.b. b.b is Bit Number 0-15BooleanD DWORDUnsigned Double WordD0-D65532 DWORD0-DWORD65532.b DWORD0.b-DWORD65532.b b.b is Bit Number 0-31DWord, Long, LBCD, FloatDI DINTSigned Double WordD10-D165532.b DINT0.b-DINT65532.b DINT0.b-DINT65532.b DINT0.b-DINT65532.b DINT0.b-DINT65532.b DINT0.b-DINT65532.b b.b is Bit Number 0-31DWord, Long, LBCD, FloatDTS7 Date_And_Time. Complex data type stored with 8 bytes as follows: 0 year, 1 month, 2 days, 3 hours, 4 minutes, 5 seconds, 6 two most significant digits of MSEC, 7 (4MSB) two least significant fight Displayed as StringREALO-REAL65532FloatTODS7 Time_Of_Day.StringSTRINGO.nSTRING65532.n n. n is string length. OCn<2 210.	INT		INTO-INT65534		
Interpretation Interpretation Boolean Demote Unsigned Double Word D0-D65532 DWORD0-DWORD65532 DWord, Long, LBCD, Float DWORD Double Word D0-D65532 DWORD0-DWORD65532.b .b is Bit Number 0-31 Boolean DI DINT Signed Double Word D10-D165532 DIU-0-D165532.b .b is Bit Number 0-31 DWord, Long, LBCD, Float DI DI .b -D165532.b .b is Bit Number 0-31 DWord, Long, LBCD, Float Boolean DT S7 Date_And_Time. DT0-D165532.b .b is Bit Number 0-31 Boolean DT S7 Date_And_Time. DT0-D165532.b .b is Bit Number 0-31 Boolean DT S7 Date_And_Time. DT0-D165532.b .b is Bit Number 0-31 String DT S7 Date_And_Time. DT0-D165532.b .b is Bit Number 0-31 String DT S7 Date_And_Time. DT0-D165532.b .b is Bit Number 0-31 String Displayed as tring format "m/d/y h:mm:ss <am pm="">=" with range "11/1/1900 0:00:00 MM" to "12/31/2089 23:59:59 PM". Read Only. Float REAL IEEE Float REAL0-REAL65532 Float String S7 String STRING0.n-STRING65532.n .n is string length. 0<rk 210.<="" <="" td=""> String TOD String format "h:m:s.mmm" with range "0:0:0.0" to "23:59:59.999". String Read Only Read Only Read Only String</rk></am>					
INIO.D-INI65334.6. b.b. is Bit Number 0-15 D Unsigned D0-065532 DWORD Double Word D0-065532 DI D0.b-055532.b. D0.b-05532.b. DI Disgned D00-0165532 DI Double Word D10-0165532.b. DINT Double Word D10-D165532.b. DINT Double Word D10-D165532.b. DINT Signed D10.b-D165532.b. DINT S7 Date_And_Time. DT0-DT65528 Complex data type stored with 8 bytes as follows: 0 year, 1 month, 2 days, 3 hours, 4 minutes, 5 seconds, 6 two most significant digits of MSEC, 7 (4MSB) two least significant fight and were (1=Sunday). Displayed as string format "m/dy h:mm:ss < AM/PM> " with range "11/1990 0:00:00 AM" to "12/31/2089 23:59:59 PM". REAL IEEE Float String S7 String S7 Time_Of_Day. TOD String length. 0 0<6:0:00 two "23:59:59.999".			10.b-165534.b	Boolean	
D Unsigned Double Word D0-D65532 DWORD0-DWORD65532.b DWORD0-DWORD65532.b DWORD0.b-D065532.b DWORD0.b-D065532.b DWORD0.b-DWORD65532.b DWORD0.b-DWORD65532.b DWORD0.b-DWORD65532.b DWORD0.b-DWORD65532.b DWOrd, Long, LBCD, Float DI DI DINT Signed Double Word DI0-D165532 DI0.b-D165532.b DI0.b-D165532.b DWOrd, Long, LBCD, Float Boolean DI DI DIT S7 Date_And_Time. DI0-D165532.b DINT0.b-D1NT65532.b DWORD0.b-1055532.b Boolean DT S7 Date_And_Time. DT0-DT65528 String Complex data type stored with 8 bytes as follows: 0 year, 1 month, 2 days, 3 hours, 4 minutes, 5 seconds, 6 two most significant digits of MSEC, 7 (4LSB) day of week (1=Sunday). String String Displayed as string format "m/d/y h:mm:ss <am pm="">" with range "11/1/1900 0:0:0:0:0 MM" to "12/31/2089 23:59:59 PM". Float String REAL IEEE Float REAL0-REAL65532 Float String TOD S7 Time_Of_Day. String STRING0.n-STRING65532.n .n is string length. 0 <n <="210.</td"> String String TOD Stored as DWORD, representing milliseconds since midnight. Displayed as String format "h:m:s.mmm" with range "0:0:0.0" to "23:59:59.999". String String</n></am>			IN10.D-IN165534.D		
D Unsigned D0-D65532 DWORD DWORD Double Word DWORD-DWORD65532.b Boolean DI Signed D0.b-D65532.b DWORD.0565532 Boolean DINT Signed D10-D1055532.b DWORD.0565532 Boolean DINT Double Word D10-D1055532.b DWORD.5532.b Boolean DINT S7 Date_And_Time. D10.b-D165532.b Boolean DT S7 Date_And_Time. DT0-DT65528 String O year, 1 month, 2 days, 3 hours, 4 minutes, 5 seconds, 6 two most significant digits of MSEC, 7 (4LSB) two least significant digits of MSEC, 7 (4LSB) day of week (1=Sunday). D10-D165532 Float REAL IEEE Float REAL0-REAL65532 Float String S7 String STRING0.n STRING65532.n .n is string length. 0 <rr> O S7 Time_Of_Day. TOD0-T0D65532 String TOD S7 Time_Of_Day. TOD0-T0D65532 String Stored as DWORD, representing milliseconds since midnight. Displayed as String format *1:m:s.mmm" with range *0:0:0.0" to *23:59:59.999". String</rr>			.D IS BIT NUMBER 0-15		
DWORD Double Word DWORDO-DWORD65532 Boolean DI Signed DUOD05532.b. .b is Bit Number 0-31 Boolean DI Divertify Double Word Divertify DINT Double Word Divertify Boolean DI Divertify Divertify Boolean DI Divertify Divertify Boolean DIT S7 Date_And_Time. Divertify Boolean DT S7 Date_And_Time. DT-DT65528 String Oppertify String DT-DT65528 String 0 year, 1 month, 2 days, 3 hours, 4 minutes, 5 seconds, 6 two most significant digits of MSEC, 7 (4MSB) two least significant digits of MSEC, 7	D	Unsigned	D0-D65532	DWord , Long, LBCD, Float	
DI DINTSigned Double WordD0.b-D65532.b DWORD0.b-DWORD65532.b .b is Bit Number 0-31BooleanDI DINTSigned Double WordDI.b-D165532 DINTO.b-DINT65532.b DINTO.b-DINT65532.b DINTO.b-DINT65532.b DINTO.b-DINT65532.b DINTO.b-DINT65532.b DINTO.b-DINT65532.bBooleanDTS7 Date_And_Time.DT0-DT65528StringDTS7 Date_And_Time.DT0-DT65528StringO year, 1 month, 2 days, 3 hours, 4 minutes, 5 seconds, 6 two most significant digits of MSEC, 7 (4MSB) two least significant digits of MSEC, 7 (4LSB) day of week (1=Sunday).Displayed as string format "m/d/y h.mm.ss <am pm="">" with range "1/1/1990 0:00:00 AM" to "12/31/2089 23:59:59 PM".FloatREALIEEE FloatREAL0-REAL65532FloatStringS7 StringSTRING0.n-STRING65532.n .o<sr 210.<="" <="" td="">StringTODS7 Time_Of_Day.TOD0-TOD65532StringStored as DWORD, representing milliseconds since midnight. Displayed as String format "h:m:s.mmm" with range "0:0:0:0' to "23:59:59.999".FloatBead Only.Bead Only.StringString</sr></am>	DWORD	Double Word	DWORD0-DWORD65532		
DI DU, D-D6532.b Boolean DI Signed DIO-DI65532.b DWord, Long, LBCD, Float DINT Double Word DIO-DI65532.b Boolean DI DiNT Double Word DIO-DI65532.b Boolean DIT S7 Date_And_Time. DTO-DT65528 String DT S7 Date_And_Time. DTO-DT65528 String Oyear, 1 month, 2 days, 3 hours, 4 minutes, 5 seconds, 6 two most significant digits of MSEC, 7 (4MSB) two least sintermose sintermose significant digits of					
DWORD0.6-DWORD0532.b DI bis Bit Number 0-31 DI Double Word DINT Dio-Di65532 DINT Dio-Di65532.b DI Di0-Di65532.b DINT0.b-DINT65532.b Boolean DIT S7 Date_And_Time. DT S7 Date_And_Time. Complex data type stored with 8 bytes as follows: DT0-DT65532.b 0 year, 1 month, 2 days, 3 hours, 4 minutes, 5 seconds, 6 two most significant digits of MSEC, 7 (4MSB) two least significant digits of MSEC, 7 (4MSB) two least significant digits of MSEC, 7 (4MSB) two least significant digits of MSEC, 7 (4LSB) day of week (1=Sunday). Displayed as string format "m/dy h:mm:ss <am pm=""> " with range "1/1/1990 ::00:00 AM" to "12/31/2089 23:59:59 PM". Read Only. REAL TCD S7 Time_Of_Day. Stored as DWORD, representing milliseconds since midnight. Displayed as string format "h:m:s.mmm" with range "0:00:00" to "23:59:59.999". Read Only.</am>			D0.b-D65532.b	Boolean	
DI DI DINT Signed Double Word DIO-DI65532 DINTO-DINT65532 DWord, Long, LBCD, Float DINT Double Word DIO-DI65532.b DINTO.b-DINT65532.b DINTO.b-DINT65532.b b is Bit Number 0-31 Boolean DT S7 Date_And_Time. DT0-DT65528 String Complex data type stored with 8 bytes as follows: DT0-DT65528 String 0 year, 1 month, 2 days, 3 hours, 4 minutes, 5 seconds, 6 two most significant digits of MSEC, 7 (4MSB) two least significant digits of MSEC, 7 (attributes of MSEC, 7 (4LSB) day of week (1=Sunday). Float Displayed as string format "m/d/y h:mm:ss <am pm="">" with range "1/1/1900 :00:00 AM" to "12/31/2089 23:59:59 PM". Read Only. REAL0-REAL65532 Float REAL IEEE Float REAL0-REAL65532 String String S7 String STRING0.n-STRING65532.n .n is string length. Orn<<td>String String TOD S7 Time_Of_Day. TOD-T0D65532 String Stored as DWORD, representing milliseconds since midnight. Displayed as String format "h:m:s.mmm" with range "0:0:0.0" to "23:59:59.999". String</td><td></td><td></td><td>DWORD0.b-DWORD65532.b</td><td></td></am>	String String TOD S7 Time_Of_Day. TOD-T0D65532 String Stored as DWORD, representing milliseconds since midnight. Displayed as String format "h:m:s.mmm" with range "0:0:0.0" to "23:59:59.999". String			DWORD0.b-DWORD65532.b	
DI DINT Signed Double Word DI0-DI55532 DINT0-DDINT65532.b DINT0.b-DINT65532.b DINT0.b-DINT65532.b DINT0.b-DINT65532.b DWord, Long, LBCD, Float DT S7 Date_And_Time. DI0-DI655528 Boolean DT S7 Date_And_Time. DT0-DT65528 String O year, 1 month, 2 days, 3 hours, 4 minutes, 5 seconds, 6 two most significant digits of MSEC, 7 (4LSB) two least significant digits of MSEC, 7 (4LSB) day of week (1=Sunday). DT0-DT65532 Float Displayed as string format "m/d/y h:mm:ss <am pm="">" with range "12/131/2089 23:59:59 PM". ReAL0-REAL65532 Float REAL IEEE Float REAL0-REAL65532 String String S7 String STRINGO.n-STRING65532.n .o is string length. 0<n<= 210.<="" td=""> String TOD S7 Time_Of_Day. TOD0-TOD65532 String Stored as DWORD, representing milliseconds since midnight. Displayed as String format "h:m:s.mmm" with range "0:0:0:0" to "23:59:59.999". String</n<=></am>			.b is Bit Number 0-31		
DINT Double Word DINTO - DINT65532 DI Di 0.b-DIG5532.b DINTO.b-DINT65532.b b is Bit Number 0-31 Boolean DT S7 Date_And_Time. DT0-DT65528 String Complex data type stored with 8 bytes as follows: 0 year, 1 month, 2 days, 3 hours, 4 minutes, 5 seconds, 6 two most significant digits of MSEC, 7 (4LSB) day of week (1=Sunday). Displayed as string format "m/d/y h:mm:ss <am pm="">" with range "1/1/1990 0:00:00 AM" to "12/31/2089 23:59:59 PM". Float REAL IEEE Float REAL0-REAL65532 Float String S7 String STRINGO.n-STRING65532.n .n is string length. 0 <n<= 210.<="" td=""> String TOD S7 Time_0f_Day. TOD0-T0D65532 String Stored as DWORD, representing milliseconds since midnight. Displayed as String format ":m:s.mm" with range "0:0:0.0" to "23:59:59.9999". String</n<=></am>	DI	Signed	DI0-DI65532	DWord, Long , LBCD, Float	
DIO.b-DI05532.b DINT0.b-DINT65532.b b is Bit Number 0-31BooleanDTS7 Date_And_Time.DT0-DT65528StringOrgex, 1 month, 2 days, 3 hours, 4 minutes, 5 seconds, 6 two most significant digits of MSEC, 7 (4MSB) two least significant digits of MSEC, 7 (4MSB) two least significant digits of MSEC, 7 (4LSB) day of week (1=Sunday).StringStringDisplayed as string format "m/d/y h:mm:ss <am pm="">" with range "12/31/2089 23:59:59 PM".REALREAL0-REAL65532FloatREALIEEE FloatSTRING0.n-STRING65532.n .n is string length. 0<n<= 210.<="" td="">StringStringTODS7 Time_Of_Day.TOD0-TOD65532StringStringStored as DWORD, representing milliseconds since midnight. Displayed as String format "h:m:s.mmm" with range "0:0:0.0" to "23:59:59.999".TOD0-TOD65532String</n<=></am>	DINT	Double Word	DINT0-DINT65532		
DID.b-D165532.b DINT6.b-DINT65532.b b.b is Bit Number 0-31BooleanDTS7 Date_And_Time.DT0-DT65528StringComplex data type stored with 8 bytes as follows:0 year, 1 month, 2 days, 3 hours, 4 minutes, 5 seconds, 6 two most significant digits of MSEC, 7 (4MSB) two least significant digits of MSEC, 7 (4LSB) day of week (1=Sunday).StringDisplayed as string format "m/d/y h:mm:ss <am pm="">" with range "1/1/1990 0:00:00 AM" to "12/31/2089 23:59:59 PM".FloatREALIEEE FloatREAL0-REAL65532FloatStringS7 StringSTRING0.n-STRING65532.n .n is string length. 0<n<= 210.<="" td="">StringTODS7 Time_Of_Day.TOD0-TOD65532StringStored as DWORD, representing milliseconds since midnight. Displayed as String format "h:m:s.mmm" with range "0:01:0.0" to "23:59:59.999".String</n<=></am>					
DINTO.b-DINT65532.b .b is Bit Number 0-31 DT S7 Date_And_Time. DT0-DT65528 Complex data type stored with 8 bytes as follows: 0 year, 1 month, 2 days, 3 hours, 4 minutes, 5 seconds, 6 two most significant digits of MSEC, 7 (4MSB) two least significant digits of MSEC, 7 (4LSB) day of week (1=Sunday). String Displayed as string format "m/d/y h:mm:ss <am pm="">" with range "1/1/1990 0:00:00 AM" to "12/31/2089 23:59:59 PM". Float REAL IEEE Float REAL0-REAL65532 Float String S7 String STRING0.n-STRING65532.n. o<n<= 210.<="" td=""> String TOD S7 Time_Of_Day. TOD0-TOD65532 String Stored as DWORD, representing milliseconds since midnight. Displayed as String format "h:m:s,mmm" with range "0:0:0.0" to "23:59:59.999". Bead Only.</n<=></am>			DI0.b-DI65532.b	Boolean	
DT S7 Date_And_Time. DT0-DT65528 String Oyear, 1 month, 2 days, 3 hours, 4 minutes, 5 seconds, 6 two most significant digits of MSEC, 7 (4MSB) two least significant digits of MSEC, 7 (4LSB) day of week (1=Sunday). Displayed as string format "m/d/y h:mm:ss <am pm="">" with range "1/1/1990 0:00:00 AM" to "12/31/2089 23:59:59 PM". Float REAL IEEE Float REAL0-REAL65532 Float String S7 String STRING0.n-STRING65532.n .n is string length. 0<n<= 210.<="" td=""> String TOD S7 Time_Of_Day. TOD0-TOD65532 String Stored as DWORD, representing milliseconds since midnight. Displayed as String of using "0:00.0" to "23:59:59.999". TOD0-TOD65532 String</n<=></am>			DINT0.b-DINT65532.b		
DT S7 Date_And_Time. DT0-DT65528 String Complex data type stored with 8 bytes as follows: 0 year, 1 month, 2 days, 3 hours, 4 minutes, 5 seconds, 6 two most significant digits of MSEC, 7 (4MSB) two least significant digits of MSEC, 7 (4MSB) two least significant digits of MSEC, 7 (4LSB) day of week (1=Sunday). bisplayed as string format "m/d/y h:mm:ss <am pm="">" with range "1/1/1990 0:00:00 AM" to "12/31/2089 23:59:59 PM". Float REAL IEEE Float REAL0-REAL65532 Float String S7 String STRING0.n-STRING65532.n. n. nis string length. 0<n<= 210.<="" td=""> String TOD S7 Time_Of_Day. TOD0-TOD65532 String Stored as DWORD, representing milliseconds since midnight. Displayed as String format "h:m:s.mmm" with range "0:0:0.0" to "23:59:59.999". TOD0-TOD65532 String</n<=></am>			.b is Bit Number 0-31		
Complex data type stored with 8 bytes as follows: O year, 1 month, 2 days, 3 hours, 4 minutes, 5 seconds, 6 two most significant digits of MSEC, 7 (4MSB) two least significant digits of MSEC, 7 (4LSB) day of week (1=Sunday). Displayed as string format "m/d/y h:mm:ss < AM/PN>" with range "1/1/1990 0:00 AM" to "12/31/2089 23:59:59 PM". REAL IEEE Float REALO-REAL65532 Float String S7 String STRING0.n-STRING65532.n. n: is string length. 0 <n <="210.</td"> String TOD S7 Time_Of_Day. TOD0-TOD65532 String String or S7 Time_of_Day. TOD0-TOD65532 String Read Only. Bastring format "minit, bisplayed as String format "minit, minit, mm" with range "0:0:0.0" to "23:59:59.999". Float</n>	DT	S7 Date_And_Time.	DT0-DT65528	String	
Complex data type stored with 8 bytes as follows:Complex data type stored with 8 bytes as follows:0 year, 1 month, 2 days, 3 hours, 4 minutes, 5 seconds, 6 two most significant digits of MSEC, 7 (4MSB) two least significant digits of MSEC, 7 (4LSB) day of week (1=Sunday).Displayed as string format "m/d/y h:mm:ss <am pm="">" with range "1/1/1990 0:00:00 AM" to "12/31/2089 23:59:59 PM".Read Only.REALIEEE FloatStringS7 StringST StringST StringST StringST StringST String of Day.Stored as DWORD, representing milliseconds since midnight. Displayed as String format "h:m:s.mmm" with range "0:0:0.0" to "23:59:59.999".Bead Only.</am>					
as follows: 0 year, 1 month, 2 days, 3 hours, 4 minutes, 5 seconds, 6 two most significant digits of MSEC, 7 (4MSB) two least significant digits of MSEC, 7 (4LSB) day of week (1=Sunday). Image: Seconds, 6 two most significant digits of MSEC, 7 (4LSB) day of week (1=Sunday). Displayed as string format "m/d/y h:mm:ss <am pm="">" with range "1/1/1990 0:00:00 AM" to "12/31/2089 23:59:59 PM". Float Read Only. Read Only. Float String S7 String STRING0.n-STRING65532.n. n. is string length. 0<n 210.<="" <="" td=""> TOD S7 Time_Of_Day. TOD0-TOD65532 String Stored as DWORD, representing milliseconds since midnight. Displayed as String format "h:m:s.mmm" with range "0:0:0.0" to "23:59:59.999". Bead Only.</n></am>		Complex data type stored with 8 bytes			
0 year, 1 month, 2 days, 3 hours, 4 minutes, 5 seconds, 6 two most significant digits of MSEC, 7 (4MSB) two least significant digits of MSEC, 7 (4LSB) day of week (1=Sunday).Image: Seconds, 6 two most significant digits of MSEC, 7 (4LSB) day of week (1=Sunday).Displayed as string format "m/d/y h:mm:ss <am pm="">" with range "1/1/1990 0:00:00 AM" to "12/31/2089 23:59:59 PM".Image: Seconds, 6 two most significant digits of MSEC, 7 (4LSB) day of week (1=Sunday).REALIEEE FloatREAL0-REAL65532FloatString\$7 StringSTRING0.n-STRING65532.n .n is string length. 0<n <="210.</td">StringTOD\$7 Time_Of_Day.TOD0-TOD65532StringStored as DWORD, representing milliseconds since midnight. Displayed as String format "h:ms.mmm" with range "0:0:0.0" to "23:59:59.999".String</n></am>		as follows:			
0 year, 1 month, 2 days, 3 hours, 4 minutes, 5 seconds, 6 two most significant digits of MSEC, 7 (4MSB) two least significant digits of MSEC, 7 (4LSB) day of week (1=Sunday).Displayed as string format "m/d/y h:mm:ss <am pm="">" with range "1/1/1990 0:00:00 AM" to "12/31/2089 23:59:59 PM".Read Only.REALIEEE FloatStringS7 StringS7 StringS7 Time_Of_Day.Stored as DWORD, representing milliseconds since midnight. Displayed as String format "h:m:s.mmm" with range "0:0:0.0" to "23:59:59.999".Read Only.</am>					
minutes, 5 seconds, 6 two most significant digits of MSEC, 7 (4MSB) two least significant digits of MSEC, 7 (4LSB) day of week (1=Sunday).Displayed as string format "m/d/y h:mm:ss <am pm="">" with range "1/1/1990 0:00:00 AM" to "12/31/2089 23:59:59 PM".Read Only.REALIEEE FloatStringS7 StringS7 StringS7 Time_Of_Day.Stored as DWORD, representing milliseconds since midnight. Displayed as String format "h:m:s.mmm" with range "0:0:0.0" to "23:59:59.999".Read Only.</am>		0 year, 1 month, 2 days, 3 hours, 4			
significant digits of MSEC, 7 (4MSB) two least significant digits of MSEC, 7 (4LSB) day of week (1=Sunday). Ieast significant digits of MSEC, 7 (4LSB) day of week (1=Sunday). Displayed as string format "m/d/y h:mm:ss < AM/PM>" with range "1/1/1990 0:00:00 AM" to "12/31/2089 23:59:59 PM". Image: The second		minutes, 5 seconds, 6 two most			
least significant digits of MSEC, 7 (4LSB) day of week (1=Sunday).Displayed as string format "m/d/y h:mm:ss <am pm="">" with range "1/1/1990 0:00:00 AM" to "12/31/2089 23:59:59 PM".Read Only.REALIEEE FloatStringS7 StringS7 StringS7 Time_Of_Day.TODS7 Time_Of_Day.Stored as DWORD, representing milliseconds since midnight. Displayed as String format "h:m:s.mmm" with range "0:0:0.0" to "23:59:59.999".Bead Only.</am>		significant digits of MSEC, 7 (4MSB) two			
(4LSB) day of week (1=Sunday).Displayed as string format "m/d/y h:mm:ss <am pm="">" with range "1/1/1990 0:00:00 AM" to "12/31/2089 23:59:59 PM".Read Only.REALIEEE FloatStringS7 StringST StringS7 Time_Of_Day.TODS7 Time_Of_Day.Stored as DWORD, representing milliseconds since midnight. Displayed as String format "h:m:s.mmm" with range "0:0:0.0" to "23:59:59.999".Read Only</am>		least significant digits of MSEC, 7			
Displayed as string format "m/d/y h:mm:ss <am pm="">" with range "1/1/1990 0:00:00 AM" to "12/31/2089 23:59:59 PM".Image: Second String StringFloatREALIEEE FloatREAL0-REAL65532FloatStringS7 StringSTRING0.n-STRING65532.n . n is string length. 0<n<= 210.<="" td="">StringTODS7 Time_Of_Day.TOD0-TOD65532StringStored as DWORD, representing milliseconds since midnight. Displayed as String format "h:m:s.mmm" with range "0:0:0.0" to "23:59:59.999".TOD0-TOD65532String</n<=></am>		(4LSB) day of week (1=Sunday).			
Displayed as string format "m/d/y h:mm:ss < AM/PM>" with range "1/1/1990 0:00:00 AM" to "12/31/2089 23:59:59 PM".FloatRead Only.Read Only.REALIEEE FloatStringS7 StringST StringST String length. 0 <n<= 210.<="" td="">TODS7 Time_Of_Day.Stored as DWORD, representing milliseconds since midnight. Displayed as String format "h:m:s.mmm" with range "0:0:0.0" to "23:59:59.999".Read Only.</n<=>					
h:mm:ss <am pm="">" with range "1/1/1990 0:00:00 AM" to "12/31/2089 23:59:59 PM".FloatRead Only.Read Only.REALIEEE FloatStringS7 StringS7 StringSTRING0.n-STRING65532.n .n is string length. 0<n<= 210.<="" td="">TODS7 Time_Of_Day.Stored as DWORD, representing milliseconds since midnight. Displayed as String format "h:m:s.mmm" with range "0:0:0.0" to "23:59:59.999".Read Only.</n<=></am>		Displayed as string format "m/d/y			
"1/1/1990 0:00:00 AM" to "12/31/2089 23:59:59 PM".Read Only.Read Only.REALIEEE FloatREAL0-REAL65532FloatStringS7 StringSTRING0.n-STRING65532.n . n is string length. 0 <n<= 210.<="" td="">StringStringTODS7 Time_Of_Day.TOD0-TOD65532StringStored as DWORD, representing milliseconds since midnight. Displayed as String format "h:m:s.mmm" with range "0:0:0.0" to "23:59:59.999".String</n<=>		h:mm:ss <am pm="">" with range</am>			
"12/31/2089 23:59:59 PM".Read Only.REALIEEE FloatStringS7 StringS7 StringS7 Time_Of_Day.Stored as DWORD, representing milliseconds since midnight. Displayed as String format "h:m:s.mmm" with range "0:0:0.0" to "23:59:59.999".Read Only.		"1/1/1990 0:00:00 AM" to			
Read Only.Read Only.REALIEEE FloatREAL0-REAL65532FloatStringS7 StringSTRING0.n-STRING65532.n .n is string length. 0 <n<= 210.<="" td="">StringTODS7 Time_Of_Day.TOD0-TOD65532StringStored as DWORD, representing milliseconds since midnight. Displayed as String format "h:m:s.mmm" with range "0:0:0.0" to "23:59:59.999".TOD0-TOD65532Bead OnlyBead OnlyRead Only</n<=>		"12/31/2089 23:59:59 PM".			
Read Only.Read Only.REALIEEE FloatREAL0-REAL65532FloatStringS7 StringSTRING0.n-STRING65532.n .n is string length. 0 <n<= 210.<="" td="">StringTODS7 Time_Of_Day.TOD0-TOD65532StringStored as DWORD, representing milliseconds since midnight. Displayed as String format "h:m:s.mmm" with range "0:0:0.0" to "23:59:59.999".Stored as Only</n<=>					
REALIEEE FloatREAL0-REAL65532FloatString\$7 StringSTRING0.n-STRING65532.n .n is string length. 0 <n<= 210.<="" td="">StringTOD\$7 Time_Of_Day.TOD0-TOD65532StringStored as DWORD, representing milliseconds since midnight. Displayed as String format "h:m:s.mmm" with range "0:0:0.0" to "23:59:59.999".Stored as DWORD, representing Note the text of tex</n<=>		Read Only.			
String S7 String STRING0.n-STRING65532.n String .n is string length. 0 <n<= 210.<="" td=""> String TOD S7 Time_Of_Day. TOD0-TOD65532 String Stored as DWORD, representing milliseconds since midnight. Displayed as String format "h:m:s.mmm" with range "0:0:0.0" to "23:59:59.999". Bead Only Bead Only</n<=>	REAL	IEEE Float	REAL0-REAL65532	Float	
.n is string length. 0 <n<= 210.<="" td=""> TOD S7 Time_Of_Day. Stored as DWORD, representing milliseconds since midnight. Displayed as String format "h:m:s.mmm" with range "0:0:0.0" to "23:59:59.999". Bead Only</n<=>	String	S7 String	STRING0.n-STRING65532.n	String	
O <n<= 210.<="" th=""> TOD S7 Time_Of_Day. TOD0-TOD65532 Stored as DWORD, representing milliseconds since midnight. Displayed as String format "h:m:s.mmm" with range "0:0:0.0" to "23:59:59.999". String</n<=>	5		.n is string length.	-	
TOD S7 Time_Of_Day. TOD0-TOD65532 String Stored as DWORD, representing milliseconds since midnight. Displayed as String format "h:m:s.mmm" with range "0:0:0.0" to "23:59:59.999". String String			0 <n<= 210.<="" td=""><td></td></n<=>		
Stored as DWORD, representing milliseconds since midnight. Displayed as String format "h:m:s.mmm" with range "0:0:0.0" to "23:59:59.999".	TOD	S7 Time Of Day.	TOD0-TOD65532	String	
Stored as DWORD, representing milliseconds since midnight. Displayed as String format "h:m:s.mmm" with range "0:0:0.0" to "23:59:59.999". Bead Only				3	
milliseconds since midnight. Displayed as String format "h:m:s.mmm" with range "0:0:0.0" to "23:59:59.999". Bead Only		Stored as DWORD, representing			
as String format "h:m:s.mmm" with range "0:0:0.0" to "23:59:59.999". Read Only		milliseconds since midnight. Displayed			
range "0:0:0.0" to "23:59:59.999".		as String format "h;m:s.mmm" with			
Read Only		range "0:0:0.0" to "23:59:59.999"			
Read Only					
itedu olinyi		Read Only.			

*These are Raw Strings that differ in structure and usage from the STEP 7 String data type.

Note: Use caution when modifying Word, Short, DWord, and Long type as each address starts at a byte offset within the device. Therefore, Words MW0 and MW1 overlap at byte 1. Writing to MW0 will also modify the value held in MW1. Similarly, DWord, and Long types can also overlap. It is recommended that these memory types be used so that overlapping does not occur. As an example, with DWord MD0, MD4, MD8 ... and so on can be used to prevent overlapping bytes.

See Also: Examples

String Support

Raw Strings

For an address DBx, By.n @ String, String values read and written are stored at Byte offset y.

у	y+1	y+2	 y+n-1
	11	11	 11

Raw strings are null terminated. If the maximum string length is 10 and 3 characters are written, the fourth character is set to NULL, while characters 5-10 are left untouched.

String Support

The String subtype follows the STEP 7 String data type definition. The syntax for the String S7 Data Type is STRINGy.n where y is the Byte offset, and n is the maximum String length. If n is not specified, the maximum String length will be 210 characters. String values read and written are stored at Byte offset y+2 in Data Block x. The "actual string length" gets updated with every write based on the string length of the string being written.

У	y+1	y+2	y+3	y+4	 y+2+n-1
max string length (n)	actual string length				 11

Note: String Strings are NULL padded. If the maximum string length is 10 and 3 characters are written, characters 4-10 are set to NULL.

Array Support

The [rows][cols] notation is appended to an address to specify an array (e.g. MW0[2][5]). If no rows are specified, row count of 1 is assumed. Boolean Arrays and String Arrays are not supported.

For Word, Short and BCD arrays, the base address + (rows * cols * 2) cannot exceed 65536. Keep in mind that the elements of the array are words, located on a word boundary. For example, IW0[4] would return IW0, IW2, IW4, and IW6.

For Float, DWord, Long and Long BCD arrays, the base address + (rows * cols * 4) cannot exceed 65536. Keep in mind that the elements of the array are DWord, located on a DWord boundary. For example, ID0[4] will return ID0, ID4, ID8, ID12.

For all arrays, the total number of bytes being requested cannot exceed the internal block size of 212 bytes.

Timers

The Siemens TCP/IP Ethernet driver automatically scales T values based on the Siemens S5 time format. Timer data is stored as a Word in the PLC but scaled to a DWord in the driver. The value returned will already be scaled using the appropriate Siemens time base. As a result, the values are always returned as a count of milliseconds. When writing to T memory the Siemens time base will also be applied. To write a value to a timer in the controller, simply write the desired value as a count of milliseconds to the appropriate timer.

Counters

The value returned for C memory will automatically be converted to a BCD value.

Examples

S7 Data Type	Data Type	Input	Flags	DB
Х	Boolean	IX0.7	MX0.7	DB1,X0.7
В	Byte	IB0	MB0	DB1,B0
Byte		IBYTE0	MBYTE0	DB1,BYTE0
	Boolean	IB0.7 IBYTE0.7	MB0.7 MBYTE0.7	DB1,B0.7 DB1,BYTE0.7
	String	IB0.64 IBYTE0.64	MB0.64 MBYTE0.64	DB1,B0.64 DB1,BYTE0.64
	Array	IB0[2][5] IBYTE0[2][5]	MB0[2][5] MBYTE0[2][5]	DB1,B0[2][5] DB1,BYTE0[2][5]
С	Char	IC0	MC0	DB1,C0
Char		ICHAR0	MCHAR0	DB1,CHAR0
	Boolean	IC0.7 ICHAR0.7	MC0.7 MCHAR0.7	DB1,C0.7 DB1,CHAR0.7

	String	IC0.64 ICHAR0.64	MC0.64 MCHAR0.64	DB1,C0.64 DB1.CHAR0.64
	Array	IC0[10] ICHAR0[10]	MC0[10] MCHAR0[10]	DB1,C0[10] DB1,CHAR0[10]
W Word	Word	IW0 IWORD0	MW0 MWORD0	DB1,W0 DB1,WORD0
	Boolean	IW0.15 IWORD0.15	MW0.15 MWORD0.15	DB1,W0.15 DB1,WORD0.15
	Array	IW0[10] IWORD0[10]	MW0[10] MWORD0[10]	DB1,W0[10] DB1,WORD0[10]
I INT	Short	IIO IINTO	MIO MINTO	DB1,I0 DB1,INT0
	Boolean	IIO.15 IINTO.15	MI0.15 MINT0.15	DB1,I0.15 DB1,INT0.15
	Array	II0[5][2] IINT0[5][2]	MI0[5][2] MINT0[5][2]	DB1,I0[5][2] DB1,INT0[5][2]
D DWORD	DWord	ID0 IDWORD0	MD0 MDWORD0	DB1,D0 DB1,DWORD0
	Boolean	ID0.31 IDWORD0.31	MD0.31 MDWORD0.31	DB1,D0.31 DB1,DWORD0.31
	Array	ID0[10] IDWORD0[10]	MD0[10] MDWORD0[10]	DB1,D0[10] DB1,DWORD0[10]
DI DINT	Long	IDI0 IDINT0	MDI0 MDINT0	DB1,DI0 DB1,DINT0
	Boolean	IDI0.31 IDINT0.31	MDI0.31 MDINT0.31	DB1,DI0.31 DB1,DINT0.31
	Array	IDI0[4][3] IDINT0[4][3]	MDI0[4][3] MDINT0[4][3]	DB1,DI0[4][3] DB1,DINT0[4][3]
DT	String	N/A.	N/A.	DB1,DT0
REAL	Float	IREAL0	MREALO	DB1,REAL0
	Array	IREAL0[10]	MREAL0[10]	DB1,REAL0[10]
String	String	ISTRING0.10	MSTRING0.10	DB1,STRING0.10
TOD	String	N/A.	N/A.	DB1,TOD0

Applicom Direct-Link SW1000 Item Syntax

The following support for the Applicom Direct-Link SW1000 OPC Server is considered to be limited. Care must be taken as the data type for a given S7 data type/suffix may differ from the data type for the same S7 data type/suffix in the specified product. Applicom ASCII Strings are not supported by this driver. The following is intended to be a guideline for those already familiar with and/or prefer the syntax of the specified product. For preferred item syntax, refer to **Standard S7-300/400/1200 Item Syntax**.

Address Syntax

Input, Output, Peripheral, Flag Memory Types <memory type><S7 data type><address> <memory type><S7 data type><address><_row_col>

Timer and Counter Memory Types

<memory type><address>

DB Memory Type

DB<num>.<S7 data type><address>[<suffix>] DB<num>.<S7 data type><address><_row_col> Where <num> ranges from 1 to 65535.

See Also: Examples

Memory Types

Memory Type	Description	Address Range	Data Type	Access	
I	Inputs			Read/Write	
E					
Q	Outputs				
A					
PI	Peripheral Inputs	Dependent on C7 Data Tu	Read Only		
PE		(See table below)			
PQ	Peripheral Outputs	(See table below)		Read/Write	
PA					
Μ	Flag Memory			Read/Write	
F					
DB	Data Blocks			Read/Write	
Т	Timers	T0-T65535	Read/Write		
С	Counters	C0-C65535	Read/Write		
Z		Z0-Z65535			

See Also: Examples

S7 Data Types

The S7 data type is used to coerce the data type for a tag. It does not apply to Timers and Counters. The default data types are shown in **bold**. Suffixes are not required.

Data Type	Description	Address Range	Data Type	Suffix	Data Type w/ Suffix
None* DBX**	Bit	0.b-65534.b DBX0.b-DBX65534.b .b is Bit Number 0-15	Boolean		
B DBB**	Unsigned Byte	B0-B65535 DBB0-DBB65535	Byte , Char		
W DBW**	Unsigned Word	W0-W65534 DBW0-DBW65534	Word, Short, BCD		
D DBD**	Unsigned Double Word	D0-D65532 DBD0-DBD65532	DWord, Long, LBCD, Float	F	Float

*No S7 data type specified. Applies to non-DB Memory Types Only.

**Applies to DB Memory Types Only.

See Also: Examples

Suffixes

Suffix	Description	Data Type
F	32 bit IEEE floating point value	Float

Array Support

The <.array size> notation is appended to an address to specify an array (e.g. MW0.10). Boolean Arrays and String Arrays are not supported.

Examples

S7 Data Type	Data Type	Input	Flags	DB
None	Boolean	I0.7	M0.7	
DBX	Boolean			DB1.DBX0.7
В	Byte	IB0	MB0	
DBB				DB1.DBB0
	Array	IB0_2_5	MB0_2_5	
				DB1.DBB0_2_5

Siemens TCP/IP Ethernet Driver Help

W	Word	IW0	MW0	
DBW				DB1.DBW0
	Array	IW0_10	MW0_10	
				DB1.DBW0_10
D	DWord	ID0	MD0	
DBD				DB1.DBD0
	Float (F)			
				DB1.DBD0F
	Array	ID0_4_3	MD0_4_3	
				DB1.DBD0F_4_3

Note: All brand and product names are trademarks, registered trademarks or service marks of their respective holders.

INAT OPC Server TCPIPH1 Item Syntax

Important: The following support for the INAT OPC Server TCPIPH1 (V1.22 and up) is considered to be limited. Care must be taken as the data type for a given S7 data type/suffix may differ from the data type for the same S7 data type/suffix in the specified product. S7 data type and suffixes not included below are not supported by this driver. The following is intended to be a guideline for those already familiar with and/or prefer the syntax of the specified product. For preferred item syntax, refer to **Standard S7-300/400/1200 Item Syntax**.

Address Syntax

Input, Output, Peripheral, Flag Memory Types

<memory type><S7 data type><address>[<suffix>] <memory type><S7 data type><address><.string length>* <memory type><S7 data type><address><.array size>[<suffix>]

Timer and Counter Memory Types

<memory type><address>

DB Memory Type

DB<num>.<S7 data type><address>[<suffix>] DB<num>.<S7 data type><address><.string length>* DB<num>.<S7 data type><address><.array size>[<suffix>]

Where <num> ranges from 1 to 65535.

*Applies to S7 data types that support String.

See Also: Examples

Memory Types

Memory Type	Description	Address Range	Data Type	Access
I	Inputs			Read/Write
Q A	Outputs	_	Read/Write	
PI PE	Peripheral Inputs	Dependent on S7 Data Type.		Read Only
PQ PA	Peripheral Outputs	(See table below)	Read/Write	
M F	Flag Memory		Read/Write	
DB	Data Blocks	1		Read/Write
Т	Timers	T0-T65535 DWord, Long		Read/Write
C Z	Counters	C0-C65535 Z0-Z65535	Word, Short	Read/Write

See Also: Examples

S7 Data Types

The S7 data type is used to coerce the data type for a tag. It does not apply to Timers and Counters. Default data types are shown in **bold**. A suffix is not required.

Data Type	Description	Address Range	Data Type	Suffix	Data Type w/ Suffix
Х	Bit	X0.b-X65534.b	Boolean		
		.b is Bit Number 0-15			
В	Unsigned Byte	B0-B65535	Byte , Char	KF	Char
Byte		BYTE0-BYTE65535			
W	Unsigned Word	W0-W65534	Word, Short, BCD	BCD	BCD
Word		WORD0-WORD65534		KF	Short
Ι	Signed Word	10-165534	Word, Short , BCD	BCD	BCD
INT		INT0-INT65534			
D	Unsigned	D0-D65532	DWord, Long,	BCD	LBCD
DWORD	Double Word	DWORD0-DWORD65532	LBCD, Float	IEEE	Float
				KF	Long
DI	Signed	DI0-DI65532	DWord, Long ,	BCD	LBCD
DINT	Double Word	DINT0-DINT65532	LBCD, Float	IEEE	Float
R	IEEE Float	R0-R65532	Float		
REAL		REAL0-REAL65532			
G	S7 String	G0.n-G65532.n	String	1	
String		STRING0.n-STRING65532.n			
		.n is string length.			
		0 <n<= 210.<="" td=""><td></td><td></td><td></td></n<=>			

See Also: Examples

Suffixes

Suffix	Description	Data Type
BCD	Two byte packed BCD for Word references Value range is0-9999	BCD
	Four byte packed BCD for DWord references Value range is 0-99999999	LBCD
IEEE	32 bit IEEE floating point value	Float
KF	Signed	Char
		Short
		Long

Array Support

The <.array size> notation is appended to an address to specify an array (e.g. MW0.10). Boolean Arrays and String Arrays are not supported.

Examples

S7 Data Type	Data Type	Input	Flags	DB
Х	Boolean	IX0.7	MX0.7	DB1.X0.7
В	Byte	IB0	MB0	DB1.B0
Byte		IBYTE0	MBYTE0	DB1.BYTE0
	Char (KF)	IB0KF	MB0KF	DB1.B0KF
		IBYTE0KF	MBYTE0KF	DB1.BYTE0KF
	Array	IB0KF.10	MB0KF.10	DB1.B0KF.10
		IBYTE0KF.10	MBYTE0KF.10	DB1.BYTE0KF.10
W	Word	IWO	MWO	DB1.W0
Word		IWORD0	MWORD0	DB1.WORD0
	BCD (BCD)	IW0BCD	MW0BCD	DB1.W0BCD
		IWORD0BCD	MWORD0BCD	DB1.WORD0BCD
	Short (KF)	IW0KF	MW0KF	DB1.W0KF
		IWORD0KF	MWORD0KF	DB1.WORD0KF

	Array	IW0BCD.10	MW0BCD.10	DB1.W0BCD.10
		IWORD0BCD.10	MWORD0BCD.10	DB1.WORD0BCD.10
I	Short	IIO	MIO	DB1.I0
INT		IINTO	MINTO	DB1.INT0
	BCD (BCD)	II0BCD	MI0BCD	DB1.I0BCD
		IINTOBCD	MINTOBCD	DB1.INT0BCD
	Array	II0.10	MI0.10	DB1.I0.10
		IINT0.10	MINT0.10	DB1.INT0.10
D	DWord	ID0	MD0	DB1.D0
DWORD		IDWORD0	MDWORD0	DB1.DWORD0
	LBCD (BCD)	ID0BCD	MD0BCD	DB1.D0BCD
		IDWORD0BCD	MDWORD0BCD	DB1.DWORD0BCD
	Float (IEEE)	ID0IEEE	MD0IEEE	DB1.D0IEEE
		IDWORD0IEEE	MDWORD0IEEE	DB1.DWORD0IEEE
	Long (KF)	ID0KF	MD0KF	DB1.D0KF
		IDWORD0KF	MDWORD0KF	DB1.DWORD0KF
	Array	ID0IEEE.10	MD0IEEE.10	DB1.D0IEEE.10
		IDWORD0IEEE.10	MDWORD0IEEE.10	DB1.DWORD0IEEE.10
DI	Long	IDI0	MDI0	DB1.DI0
DINT		IDINT0	MDINT0	DB1.DINT0
	LBCD (BCD)	IDI0BCD	MDI0BCD	DB1.DI0BCD
		IDINTOBCD	MDINTOBCD	DB1.DINT0BCD
	Float (IEEE)	IDI0IEEE	MDIOIEEE	DB1.DI0IEEE
		IDINTOIEEE	MDINTOIEEE	DB1.DINT0IEEE
	Array	IDI0BCD.10	MDI0BCD.10	DB1.DI0BCD.10
		IDINT0BCD.10	MDINT0BCD.10	DB1.DINT0BCD.10
R	Float	IRO	MR0	DB1.R0
REAL		IREAL0	MREAL0	DB1.REAL0
	Array	IR0.10	MR0.10	DB1.R0.10
		IREAL0.10	MREAL0.10	DB1.REAL0.10
G	String	IG0.10	MG0.10	DB1.G0.10
String	-	ISTRING0.10	MSTRING0.10	DB1.STRING0.10

Note: All brand and product names are trademarks, registered trademarks or service marks of their respective holders.

Siemens Simatic Net Item Syntax

Important: The following support for the Siemens Simatic Net OPC Server is considered to be limited. Care must be taken as the data type for a given S7 data type may differ from the data type for the same S7 data type in the specified product. S7 data types not included below are not supported by this driver. The following is intended to be a guideline for those already familiar with and/or prefer the syntax of the specified product. For preferred item syntax, refer to **Standard S7-300/400/1200 Item Syntax**.

Address Syntax

Input, Output, Peripheral, Flag Memory Types

<memory type><S7 data type><address> <memory type><S7 data type><address><.string length> <memory type><S7 data type><address><,array size>

Timer and Counter Memory Types

<memory type><address>

DB Memory Type

DB<num>,<S7 data type><address> DB<num>,<S7 data type><address><.string length>* DB<num>,<S7 data type><address><,array size>

Where <num> ranges from 1 to 65535.

*Applies to S7 data types that support String.

See Also: Examples

Memory Types

Memory Type	Description	Address Range	Data Type	Access
I	Inputs			Read/Write
E		Dependent on S7 Data Type. (See table below)		
Q	Outputs			Read/Write
А				
PI	Peripheral Inputs			Read Only
PE				
PQ	Peripheral Outputs			Read/Write
PA				
Μ	Flag Memory			Read/Write
F				
DB	Data Blocks			Read/Write
Т	Timers	T0-T65535 DWord, Long		Read/Write
С	Counters	C0-C65535	Word, Short	Read/Write
Z		Z0-Z65535		

See Also: Examples

S7 Data Types

The S7 data type is used to coerce the data type for a tag. It does not apply to Timers and Counters. The default data types are shown in **bold**.

Data Type	Description	Address Range	Data Type
Х	Bit	X0.b-X65534.b	Boolean
		.D IS BIL NUMBER 0-15	
В	Unsigned Byte	B0-B65535	Byte, Char
Byte		BYTE0-BYTE65535	
Char	Signed Byte	CHAR0-CHAR65535	Byte, Char
W	Unsigned Word	W0-W65534	Word, Short, BCD
Word		WORD0-WORD65534	
INT	Signed Word	INT0-INT65534	Word, Short , BCD
D	Unsigned	D0-D65532	DWord, Long, LBCD, Float
DWORD	Double Word	DWORD0-DWORD65532	
DINT	Signed	DINT0-DINT65532	DWord, Long , LBCD, Float
	Double Word		
REAL	IEEE Float	REAL0-REAL65532	Float
String	S7 String	STRING0.n-STRING65532.n	String
		.n is string length.	
		0 <n<= 210.<="" td=""><td></td></n<=>	

See Also: Examples

Array Support

The <.array size> notation is appended to an address to specify an array (e.g. MW0.10). Boolean Arrays and String Arrays are not supported.

Examples

S7 Data Type	Data Type	Input	Flags	DB
Х	Boolean	IX0.7	MX0.7	DB1,X0.7
В	Byte	IB0	MB0	DB1,B0

Byte		IBYTE0	MBYTE0	DB1,BYTE0
	Array	IB0,10	MB0,10	DB1,B0,10
	,	IBYTE0.10	MBYTE0.10	DB1.BYTE0.10
Char	Char		МСНАРО	
Cital	Cilai	ICHARO	MCHARU	DB1,CHARU
	Array	ICHARU, IU	MCHARU,10	DB1,CHAR0,10
W	Word	IWO	MW0	DB1,W0
Word		IWORD0	MWORD0	DB1,WORD0
	Arrav	IW0.10	MW0.10	DB1.W0.10
	,	IWORDO 10	MWORDO 10	DB1 WORD0 10
	Chart		MINTO	
	Snort	111110	MINTO	DB1,INTU
	Array	IINTO,10	MINTO,10	DB1,INT0,10
D	DWord	ID0	MD0	DB1,D0
DWORD		IDWORD0	MDWORD0	DB1,DWORD0
				,
	Array	ID0 10	MD0 10	DB1 D0 10
	7.11.0.9			
DINIT			MDW0RD0,10	
DINI	Long	IDINTO	MDINTO	DB1,DINT0
	Array	IDINT0,10	MDINT0,10	DB1,DINT0,10
REAL	Float	IREAL0	MREAL0	DB1,REAL0
	Array	IRFAL0.10	MREAL0.10	DB1.REAL0.10
Chuin a	Chuing			
String	String	ISTRINGU.10	MSTRINGU.10	DB1,STRINGU.10

Note: All brand and product names are trademarks, registered trademarks or service marks of their respective holders.

Siemens STEP 7 Item Syntax

Important: The following support for Siemens STEP 7 Variable Table (VAT) syntax is considered to be limited. Care must be taken as the data type for a given S7 data type/suffix may differ from the data type for the same S7 data type/suffix in the specified product. S7 data types not included below are not supported by this driver. The following is intended to be a guideline for those already familiar with and/or prefer the syntax of the specified product. For preferred item syntax, refer to Standard S7-300/400/1200 Item Syntax.

Address Syntax

Input, Output, Peripheral, Flag Memory Types

<memory type><S7 data type><address>

Timer and Counter Memory Types

<memory type><address>

DB Memory Type

DB<num>.<S7 data type><address>

Where <num> ranges from 1 to 65535.

See Also: Examples _

...

memory Types				
Memory Type	Description	Address Range	Data Type	Access

I E	Inputs			Read/Write
Q A	Outputs	Dependent on S7 Data Type. (See table below)		Read/Write
PI PE	Peripheral Inputs			Read Only
PQ PA	Peripheral Outputs			Read/Write
M F	Flag Memory			Read/Write
DB	Data Blocks			Read/Write
Т	Timers	T0-T65535	DWord, Long	Read/Write
C Z	Counters	C0-C65535 Z0-Z65535	Word, Short	Read/Write

See Also: Examples

Accessing Structured Elements in STEP 7

For the Siemens S7-1200 model, STEP 7 accesses the Counter and Timer structured elements as complete whole structures instead of individual tags. For more information on the element offset and its server address equivalent, refer to the tables below.

Counters

Element	Data Type	Offset	Server Address Equivalent
Count_UP	Boolean	0.0	DB1,C00.0
Count_Down	Boolean	0.1	DB1,C00.1
Reset	Boolean	0.2	DB1,C00.2
Load	Boolean	0.3	DB1,C00.3
Q_UP	Boolean	0.4	DB1,C00.4
Q_Down	Boolean	0.5	DB1,C00.5
PAD	Byte	1.0	DB1,B1
Preset_Value	Short	2.0	DB1,I2
Count_Value	Short	4.0	DB1,I4

Timers

Element	Data Type	Offset	Server Address Equivalent
Start	DWord	0.0	DB1,D0
Preset	DWord	4.0	DB1,D4
Elapsed	DWord	8.0	DB1,D8
Running	Bool	12.0	DB1,DBX12.0
IN	Bool	12.1	DB1,DBX12.1
Q	Bool	12.2	DB1,DBX12.2
PAD	Byte	13.0	DB1,DBB13
PAD_2	Byte	14.0	DB1,DBB14
PAD_3	Byte	15.0	DB1,DBB15

Note: For more information, refer to Standard S7-300/400/1200 Item Syntax.

S7 Data Types

The S7 data type is used to coerce the data type for a tag. It does not apply to Counters and Timers. The default data types are shown in **bold**.

Data Type	Description	Address Range	Data Type	Suffix	Data Type w/ Suffix
None*	Bit	0.b-65534.b	Boolean		
DBX**		DBX0.b-DBX65534.b			
		.b is Bit Number 0-15			

B DBB**	Unsigned Byte	B0-B65535 DBB0-DBB65535	Byte, Char		
W DBW**	Unsigned Word	W0-W65534 DBW0-DBW65534	Word, Short, BCD		
D DBD**	Unsigned Double Word	D0-D65532 DBD0-DBD65532	DWord, Long, LBCD, Float	F	Float

*No S7 data type specified. Applies to non-DB Memory Types Only. **Applies to DB Memory Types Only.

See Also: Examples

Examples

S7 Data Type	Data Type	Input	Flags	DB
None	Boolean	I0.7	M0.7	
DBX				DB1.DBX0.7
В	Byte	IB0	MB0	
DBB				DB1.DBB0
W	Word	IW0	MW0	
DBW				DB1.DBW0
D	DWord	ID0	MD0	
DBD				DB1.DBD0

Note: All brand and product names are trademarks, registered trademarks or service marks of their respective holders.

Softing S7/S5 OPC Server Item Syntax

Important: The following support for the Softing S7/S5 OPC Server is considered to be limited. Care must be taken as the data type for a given S7 data type/suffix may differ from the data type for the same S7 data type/suffix in the specified product. The following is intended to be a guideline for those already familiar with and/or prefer the syntax of the specified product. For preferred item syntax, refer to **Standard S7-300/400/1200 Item Syntax**.

Address Syntax

Input, Output, Peripheral, Flag Memory Types <memory type><S7 data type><address>:[<suffix>]

Timer and Counter Memory Types

<memory type><address>

DB Memory Type

DB<num>.<S7 data type><address>:[<suffix>]

Where <num> ranges from 1 to 65535.

See Also: Examples

Memory Types

Memory Type	Description	Address Range	Data Type	Access
I E	Inputs			Read/Write
Q A	Outputs		Read/Write	
PI PE	Peripheral Inputs	Dependent on S7 Data Ty	Read Only	
PQ PA	Peripheral Outputs		Read/Write	
M F	Flag Memory		Read/Write	
DB	Data Blocks		Read/Write	
Т	Timers	T0-T65535	DWord, Long	Read/Write
С	Counters	C0-C65535	Word, Short	Read/Write

55

Ζ

Z0-Z65535

See Also: Examples

S7 Data Types

The S7 data type is used to coerce the data type for a tag. It does not apply to Timers and Counters. The default data types are shown in **bold**. Suffix is not required.

Data Type	Description	Address Range	Data Type	Suffix	Data Type w/ Suffix
None* DBX**	Bit	0.b-65534.b DBX0.b-DBX65534.b .b is Bit Number 0-15	Boolean		
В	Unsigned Byte	B0-B65535	Byte , Char	Byte	Byte
DBB**		DBB0-DBB65535		Char	Char
				String	String
W	Unsigned Word	W0-W65534	Word, Short, BCD	Word	Word
DBW**		DBW0-DBW65534		INT	INT
				BCD	BCD
D	Unsigned	D0-D65532	DWord, Long, LBCD, Float	DWord	DWord
DBD**	Double Word	DBD0-DBD65532		DINT	DINT
				BCD	BCD
				REAL	REAL

*No S7 data type specified. Applies to non-DB Memory Types Only. **Applies to DB Memory Types Only.

See Also: Examples

Suffixes

Suffix	Description	Data Type
Byte	Unsigned Byte	Byte
Char	Signed Byte	Char
Word	Unsigned Word	Word
INT	Signed Word	Short
DWORD	Unsigned DWord	DWord
DINT	Signed DWord	Long
BCD	Two byte packed BCD for Word references Value range is 0-9999 Four byte packed BCD for DWord references Value range is 0-99999999	BCD LBCD
REAL	32 bit IEEE floating point value	Float
String	S7 String	String

Examples

S7 Data Type	Data Type	Input	Flags	DB
None	Boolean	I0.7	M0.7	
DBX				DB1.DBX0.7
В	Byte	IB0	MB0	
DBB				DB1.DBB0
	String (String)	IB0:String	MB0:String	
				DB1.DBB0:String
W	Word	IWO	MW0	
DBW				DB1.DBW0
	BCD (BCD)	IW0:BCD	MW0:BCD	
				DB1.DBW0:BCD
D	DWord	ID0	MD0	
DBD				DB1.DBD0

LBCD (BCD)	ID0:BCD	MD0:BCD	DB1.DBD0:BCD
Float (REAL)	ID0:REAL	MD0:REAL	 DB1.DBD0:REAL

Note: All brand and product names are trademarks, registered trademarks or service marks of their respective holders.

Legacy S7-300/400 Item Syntax

The default data types for dynamically defined tags are shown in **bold**. For preferred item syntax, refer to **Standard S7-300/400/1200 Item Syntax**.

Address Type	Range	Туре	Access
Discrete Inputs	I0.b-I65535.b*	Boolean	Read/Write
	.b is Bit Number 0-7		
	IB0-IB65535	Byte, Char, String**	Read/Write
	IW0-IW65534	Word, Short, BCD	Read/Write
	IW:KT0-IW:KT65534	DWord, Long	Read/Write
	IW:KC0-IW:KC65534	Word, Short	Read/Write
	ID0-ID65532	DWord, Long, LBCD, Float	Read/Write
Discrete Inputs	E0.b-E65535.b* .b is Bit Number 0-7	Boolean	Read/Write
	EB0-EB65535**	Byte, Char, String**	Read/Write
	EW0-EW65534	Word, Short, BCD	Read/Write
	EW:KT0-EW:KT65534	DWord, Long	Read/Write
	EW:KC0-EW:KC65534	Word, Short	Read/Write
	ED0-ED65532	DWord, Long, LBCD, Float	Read/Write
Note: I and E access the same me	mory area.		-
Discrete Outputs	Q0.b-Q65535.b* .b is Bit Number 0-7	Boolean	Read/Write
	QB0-QB65535	Byte, Char, String**	Read/Write
	QW0-QW65534	Word, Short, BCD	Read/Write
	QW:KT0-QW:KT65534	DWord, Long	Read/Write
	QW:KC0-QW:KC65534	Word, Short	Read/Write
	QD0-QD65532	DWord, Long, LBCD, Float	Read/Write
Discrete Outputs	A0.b-A65535.b* .b is Bit Number 0-7	Boolean	Read/Write
	AB0-AB65535	Byte, Char, String**	Read/Write
	AW0-AW65534	Word, Short, BCD	Read/Write
	AW:KT0-AW:KT65534	DWord, Long	Read/Write
	AW:KC0-AW:KC65534	Word, Short	Read/Write
	AD0-AD65532	DWord, Long, LBCD, Float	Read/Write

Note: Q and A access the same m	emory area.		
Peripheral Inputs	PI0.b-PI65535.b*	Boolean	Read Only
	.b is Bit Number 0-7		
	PIB0-PIB65535	Byte , Char, String**	Read Only
	PIW0-PIW65534	Word, Short, BCD	Read Only
	PIW:KT0-PIW:KT65534	DWord, Long	Read Only
	PIW:KC0-PIW:KC65534	Word, Short	Read Only
	PID0-PID65532	DWord, Long, LBCD, Float	Read Only
Peripheral Inputs	PE0.b-PE65535.b*	Boolean	Read Only
	.b is Bit Number 0-7		
	PEB0-PEB65535**	Byte , Char, String**	Read Only
	PEW0-PEW65534	Word, Short, BCD	Read Only
	PEW:KT0-PEW:KT65534	DWord, Long	Read Only
	PEW:KC0-PEW:KC65534	Word, Short	Read Only
	PED0-PED65532	DWord, Long, LBCD, Float	Read Only
Note: PI and PE access the same	nemory area.		
Peripheral Outputs	PQ0.b-PQ65535.b*	Boolean	Read/Write
	.b is Bit Number 0-7		
	PQB0-PQB65535	Byte, Char, String**	Read/Write
	PQW0-PQW65534	Word, Short, BCD	Read/Write
	PQW:KT0-PQW:KT65534	DWord, Long	Read/Write
	PQW:KC0-PQW:KC65534	Word, Short	Read/Write
	PQD0-PQD65532	DWord, Long, LBCD, Float	Read/Write
Peripheral Outputs	PA0.b-PA65535.b*	Boolean	Read/Write
	.b is Bit Number 0-7		
	PAB0-PAB65535	Byte, Char, String**	Read/Write
	PAW0-PAW65534	Word, Short, BCD	Read/Write
	PAW:KT0-PAW:KT65534	DWord, Long	Read/Write
	PAW:KC0-PAW:KC65534	Word, Short	Read/Write
	PAD0-PAD65532	DWord, Long, LBCD, Float	Read/Write
Note: PQ and PA access the same	memory area.		
Internal Memory	F0.b-F65535.b* .b is Bit Number 0-7	Boolean	Read/Write
	FB0-FB65535	Byte, Char, String**	Read/Write
	FW0-FW65534	Word, Short, BCD	Read/Write
	FW:KT0-FW:KT65534	DWord, Long	Read/Write
	FW:KC0-FW:KC65534	Word, Short	Read/Write
	FD0-FD65532	DWord, Long, LBCD, Float	Read/Write

Internal Memory (Deeleen	Deed /W/with
Internal Memory	.b is Bit Number 0-7	Boolean	Read/Write
	MB0-MB65535	Byte , Char, String**	Read/Write
	MW0-MW65534	Word, Short, BCD	Read/Write
	MW:KTO-MW:KT65534	DWord, Long	Read/Write
	MW:KC0-MW:KC65534	Word, Short	Read/Write
	MD0-MD65532	DWord , Long, LBCD, Float	Read/Write
Note: F and M access the same m	emory area.		
Data Block Boolean	DB1-N:KM0.b-KM65534.b*	Boolean	Read/Write
	1-N is Block Number .b is Bit Number 0-15		,
	Alternates		
	DB1DBX0 b-DBNDBX65534 b*	Boolean	Read/Write
	1-N is Block Number		
	.b is Bit Number 0-15		
	DB1D0 h-DBND65534 h*	Boolean	Read/Write
	1-N is Block Number	Doolean	ricula, write
	.b is Bit Number 0-15		
Data Block Left Byte	DB1-N:KI 0-KI 65535	Byte, Char, String**	Read/Write
Data Dioek Eele Dyte	1-N is Block Number	byte, chur, string	riceday write
	Alternates		
		Byte Char String**	Read/Write
	1-N is Block Number	Byte , char, string	neddy write
		Byte Char String**	Pood /Write
	1-N is Block Number	Byte, char, string	Redu/ Write
Data Plack Dight Buta		Bute Char String**	Dood /W/rito
Data block Right byte	1-N is Block Number	Byte, Char, String	Redu/ White
	Alternate		
	DB1DR0-DBNDR65534	Byte Char String**	Read/Write
	1-N is Block Number		
Data Block Unsigned Word	DB1-N:KH0-KH65534	Word Short BCD	Read/Write
	1-N is Block Number		
Data Block Signed Word	DB1-N:KE0-KE65534	Word, Short, BCD	Read/Write
	1-N is Block Number		
	Alternates		
		Word Short BCD	Read/Write
	1-N is Block Number		Redu/ Write
	DB1DW0-DBNDW65534	Word Short BCD	Read/Write
	1-N is Block Number		
Data Block Signed Long	DB1-N:KD0-KD65532	DWord Long I BCD Float	Read/Write
	1-N is Block Number	E ford, Long , Loop, Hout	
	Alternates		
		DWord Long I BCD Flost	Read /Write
	1-N is Block Number	E WOLG, LONG, LDCD, FIOAL	iteau/ write
	DB1DD0-DB1DD65532	DWord, Long, LBCD, Float	Read/Write

	1-N is Block Number		
Data Block Float	DB1-N:KG0-KG65532 1-N is Block Number	Float	Read/Write
Data Block BCD	DB1-N:BCD0-BCD65534 1-N is Block Number	Word, Short, BCD	Read/Write
Data Block S5 Timer as DB	DB1-N : KT0-KT65534 1-N is Block Number	DWord, Long	Read/Write
Data Block S5 Counter as DB	DB1-N:KC0-KC65534 1-N is Block Number	Word, Short	Read/Write
Data Block String***	DB1S0.n-DB1S65535.n* .n is string length. 0 <n<= 212.<="" td=""><td>String</td><td>Read/Write</td></n<=>	String	Read/Write
Data Block String***	DB1STRING0.n-DB1STRING65535.n* .n is string length. 0 <n<= 210.<="" td=""><td>String</td><td>Read/Write</td></n<=>	String	Read/Write
Timer Current Values****	T0-T65535*	DWord, Long	Read/Write
Counter Current Values****	C0-C65535*	Word, Short	Read/Write
Counter Current Values****	Z0-Z65535*	Word, Short	Read/Write

*These memory types/subtypes do not support arrays

**Byte memory types (i.e. MB) support Strings. The syntax for strings is <address>. <length> where 0 < length <= 212.

***For more information, refer to **Data Block** Strings.

****For more information, refer to **Timers**.

*****For more information, refer to Counters.

Note 1: All offsets for memory types I, Q, and F represent a byte starting location within the specified memory type.

Note 2: Use caution when modifying Word, Short, DWord, and Long types. For I, Q, and F each address starts at a byte offset within the device. Therefore, Words FW0 and FW1 overlap at byte 1. Writing to FW0 will also modify the value held in FW1. Similarly, DWord, and Long types can also overlap. It is recommended that these memory types be used so that overlapping does not occur. For example, with DWord, FD0, FD4, FD8 ... and so on can be used to prevent overlapping bytes.

Data Block Strings

Data Block Strings can be referenced by using S subtypes or String subtypes.

S Subtype

The syntax for the S subtype is DBxSy.n where x is the Data Block, y is the Byte offset, and n is the maximum String length. String values read and written are stored at Byte offset y in Data Block x.

у	y+1	y+2	 y+n-1
11		11	 11

S Strings are null terminated. If the maximum string length is 10 and 3 characters are written, the fourth character is set to NULL, while characters 5-10 are left untouched.

String Subtype

The String subtype follows the STEP 7 String data type definition. The syntax for the String subtype is DBxSTRINGy.n where x is the Data Block, y is the Byte offset, and n is the maximum String length. If n is not specified, the maximum String length will be 210 characters. String values read and written are stored at Byte offset y+2 in Data Block x. The first two bytes contain the maximum string length (n) and the actual string length. The actual string length gets updated with every write based on the string length of the string being written.

У	y+1	y+2	y+3	y+4	 y+2+n-1
max string length (n)	actual string length				 11

String Strings are NULL padded. If the maximum string length is 10 and 3 characters are written, characters 4-10 are set to NULL.

Arrays

All memory types/subtypes with the exception of those marked with an asterisk (*), support arrays. The syntax below are valid for declaring an array. If no rows are specified, a row count of 1 is assumed.

<address>[rows][cols] <address>.rows.cols <address>,rows,cols <address>_rows_cols

For Word, Short, BCD and "KT" arrays, the base address + (rows * cols * 2) cannot exceed 65536. Keep in mind that the elements of the array are words, located on a word boundary. For example, IW0[4] would return IW0, IW2, IW4, and IW6. "KT" subtypes fall into the 16-bit category because the data stored in the PLC is contained within a Word. For more information, refer to Timers.

For Float, DWord, Long and Long BCD arrays (excluding "KT" subtypes), the base address + (rows * cols * 4) cannot exceed 65536. Keep in mind that the elements of the array are DWord, located on a DWord boundary. For example, ID0[4] will return ID0, ID4, ID8, ID12.

For all arrays, the total number of bytes being requested cannot exceed the internal block size of 212 bytes.

KL vs. KR vs. DBB

KL and KR determine whether the left byte or right byte of the data block word is returned.

Value	8	9	А	В	С
Byte	0	1	2	3	4

The following examples are from the table above.

Example 1

DB1:KH0=0x89. DB1:KL0=0x8 DB1:KR0=0x9 DB1DBB0=0x8

Example 2

DB1:KH1=0x9A DB1:KL1=0x9 DB1:KR1=0xA DB1DBB1=0x9

Timers

The Siemens TCP/IP Ethernet driver automatically scales T and KT values based on the Siemens S5 time format. Timer data is stored as a Word in the PLC but scaled to a DWord in the driver. The value returned for either a T or KT memory type will already be scaled using the appropriate Siemens time base. As a result, the values are always returned as a count of milliseconds. When writing to T or KT memory types, the Siemens time base will also be applied. To write a value to a timer in the controller, simply write the desired value as a count of milliseconds to the appropriate timer.

Counters

The value returned for either C or KC memory type will automatically be converted to a BCD value. DB1:KH0 @ BCD=DB1:KC0 @ Word.

Examples

1. To access bit 3 of Internal Memory F20, declare an address as follows: F20.3 $\,$

2. To access Data Block 5 as word memory at byte 30, declare an address as follows: DB5:KH30 $\,$

3. To access Data Block 2 byte 20 and bit 7, declare an address as follows: DB2:KM20.7

 To access Data Block 1 as left byte memory at byte 10, declare an address as follows: DB1:KL10

5. To access Internal Memory F20 as a DWORD, declare an address as follows: FD20

6. To access Input Memory I10 as a Word, declare an address as follows:

IW10

Error Descriptions

The following error/warning messages may be generated. Click on the link for a description of the message.

Address Validation

Missing address Device address '<address>' contains a syntax error Address '<address>' is out of range for the specified device or register Data Type '<type>' is not valid for device address '<address>' Device address '<address>' is Read Only Array size is out of range for address '<address>' Array support is not available for the specified address: '<address>'

Driver Error Messages

Winsock initialization failed (OS Error=n) Winsock V1.1 or higher must be installed to use the Siemens TCP/IP Ethernet device driver

Device Status Messages

Device '<device name>' is not responding Unable to connect to device '<device name>' Unable to establish association with device '<device name>' Unable to read <block size> bytes starting at address <address> on device '<device name>' Unable to write to '<address>' on device '<device name>'

Error Codes

NetLink Errors

Error Code	Source	Description
0x00		Service could be executed without an error.
0x01	Remote Station	Timeout from remote station.
0x02	Remote Station	Resource unavailable.
0x03	Remote Station	Requested function of master is not activated within the remote station.
0x11	Remote Station	No response of the remote station.
0x12	Network	Master not into the logical token ring.
0x14	Host	Resource of the local FDL controller not available or not sufficient.
0x15	Host	The specified msg.data_cnt parameter is invalid.
0x30	Remote Station	Timeout. The requested message was accepted but no indication was sent back by the remote station.
0x39	Remote Station	Sequence fault, internal state machine error.
0x85	Host	Specified offset address out of limits or not known in the remote station.
0x86	Device	Wrong PDU coding in the MPI response of the remote station.
0x87	Host	Specified length to write or to read results in an access outside of limits.

Transport Errors

Error Code	Description
0x00	Error reason not specified.
0x01	Invalid parameter code.
0x02	Invalid TPDU type.
0x03	Invalid parameter value.

Protocol Errors

Note: Links contain the error codes for the given class.

Error Class	Description
0x00	No error.
0x81	Error in the application ID of the request.
0x82	Error in the object definition (e.g. bad data type).
0x83	No resources available.
0x84	Error in the structure of the service request.

0x85	Error in the communication equipment.
0x87	Access error.
0xD2	OVS error.
0xD4	Diagnostic error.
0xD6	Protection system error.
0xD8	BuB error.
0xEF	Layer 2 specific error.

Data Access Errors

Error Code	Description
0xFF	No error.
0x01	Hardware fault.
0x03	Illegal object access.
0x05	Invalid address (incorrect variable address).
0x06	Data type is not supported.
0x07	Invalid data size/too much data.
0x0A	Object does not exist or length error.

Address Validation

The following error/warning messages may be generated. Click on the link for a description of the message.

Address Validation

Missing address Device address '<address>' contains a syntax error Address '<address>' is out of range for the specified device or register Data Type '<type>' is not valid for device address '<address>' Device address '<address>' is Read Only Array size is out of range for address '<address>' Array support is not available for the specified address: '<address>'

Missing address

Error Type:

Warning

Possible Cause:

A tag address that has been specified dynamically has no length.

Solution:

Re-enter the address in the client application.

Device address '<address>' contains a syntax error

Error Type:

Warning

Possible Cause:

A tag address that has been specified dynamically contains one or more invalid characters.

Solution:

Re-enter the address in the client application.

Address <address>' is out of range for the specified device or register

Error Type: Warning

Possible Cause:

A tag address that has been specified dynamically references a location that is beyond the range of supported locations for the device.

Solution:

Verify the address is correct; if it is not, re-enter it in the client application.

Data Type '<type>' is not valid for device address '<address>'

Error Type:

Warning

Possible Cause:

A tag address that has been specified dynamically has been assigned an invalid data type.

Solution:

Modify the requested data type in the client application.

Device address '<address>' is Read Only

Error Type:

Warning

Possible Cause:

A tag address that has been specified dynamically has a requested access mode that is not compatible with what the device supports for that address.

Solution:

Change the access mode in the client application.

Array size is out of range for address '<address>'

Error Type:

Warning

Possible Cause:

A tag address that has been specified dynamically is requesting an array size that is too large for the address type or block size of the driver.

Solution:

Re-enter the address in the client application to specify a smaller value for the array or a different starting point.

Array Support is not available for the specified address: '<address>'

Error Type:

Warning

Possible Cause:

A tag address that has been specified dynamically contains an array reference for an address type that doesn't support arrays.

Solution:

Re-enter the address in the client application to remove the array reference or correct the address type.

Driver Error Messages

The following error/warning messages may be generated. Click on the link for a description of the message.

Driver Error Messages

Winsock initialization failed (OS Error= n)

Winsock V1.1 or higher must be installed to use the Siemens TCP/IP Ethernet device driver

Winsock initialization failed (OS Error=n)

Error Type:

Fatal

OS Error	Indication	Possible Solution
10091	Indicates that the underlying network subsystem is	Wait a few seconds and restart the driver.

	not ready for network communication.	
10067	Limit on the number of tasks supported by the	Close one or more applications that may
	Windows Sockets implementation has been	be using Winsock and restart the driver.
	reached.	

Winsock V1.1 or higher must be installed to use the Siemens TCP/IP Ethernet device driver

Error Type: Fatal

Possible Cause:

The version number of the Winsock DLL found on the system is less than 1.1.

Solution:

Upgrade Winsock to version 1.1 or higher.

Device Status Messages

The following error/warning messages may be generated. Click on the link for a description of the message.

Device Status Messages

Device '<device name>' is not responding Unable to connect to device '<device name>' Unable to establish association with device '<device name>' Unable to read <block size> bytes starting at address <address> on device '<device name>' Unable to write to '<address>' on device '<device name>'

'Device <Device name>' is not responding

Error Type:

Warning

Result

If tag was being read:

- If tag is a block tag, the entire block will be invalidated. All tags within that block will be invalidated.
- If tag is an array tag or string tag, just this tag is invalidated.

If tag was being written:

• Write operation for the given tag will not take place.

Possible Cause:

1. The connection between the device and the Host PC is broken.

- 2. The response from the device took longer to receive than the amount of time specified in the "Request Timeout" device setting.
- 3. The named device may have been assigned an incorrect IP address.
- 4. Device CPU work load is too high.

Solution:

- 1. Verify the cabling between the PC and the PLC device.
- 2. Increase the Request Timeout setting so that the entire response can be handled.
- 3. Verify the IP address given to the named device matches that of the actual device.
- 4. If this error occurs frequently, decrease the tag group scan rate to reduce the work load on the PLC CPU.
- 5. Increase the Scan Cycle Load from Communication and Scan Cycle Monitoring Time.

Unable to connect to device '<device name>'

Error Type:

Warning

Result:

1. If tag is to be read in the process of connecting and device error (transport) occurred:

- If tag is a block tag, the entire block will be deactivated. All tags within that block will be deactivated. This is signified by the postfix message "...Block Deactivated."
- If tag is an array tag or string tag, just this tag is deactivated. This is signified by the postfix message "...Tag Deactivated."

2. If tag is to be read in the process of connecting and no device errors occurred:

- If tag is a block tag, the entire block will be invalidated. All tags within that block will be invalidated.
- If tag is an array tag or string tag, just this tag is invalidated.

3. If tag is to be written in the process of connecting:

• Write operation for the given tag will not take place.

Possible Causes/Solutions:

"Frame contains errors:"

1. Incorrect TPDU response size.

- 2. Unexpected frame received.
- Incorrect response code.

3. Frame sequence is out of order.

4. Device CPU work load is too high.

Solution:

1. Cable noise may cause distortion in the frame, resulting in erroneous data. It may also cause for dropped frames. Verify the cabling between the PC and the PLC device.

2. Reduce network traffic.

3. If this error occurs frequently, increase the Request Timeout and/or Fail After attempt count.

4. If this error occurs frequently, decrease the tag group scan rate to reduce the work load on the PLC CPU.

5. Increase the "Scan Cycle Load from Communication" and "Scan Cycle Monitoring Time".

"Device returned transport error [Code=<code>]"

An RFC1006 error (ISO over TCP/IP) occurred. This is the portion of the packet that encapsulates the S7 Messaging packet.

Solution:

Contact Technical Support.

Note:

No protocol or data access errors can occur for this operation.

See Also:

Error Codes

Unable to establish association with device '<device name>'

Error Type:

Warning

Result:

1. If tag is to be read in the process of establishing an association and a device error (i.e. transport, protocol) occurred:

- If tag is a block tag, the entire block will be deactivated. All tags within that block will be deactivated. This is signified by the postfix message "...Block Deactivated."
- If tag is an array tag or string tag, just this tag is deactivated.
- This is signified by the postfix message "...Tag Deactivated."

2. If tag is to be read in the process of establishing an association and no device errors occurred.

- If tag is a block tag, the entire block will be invalidated. All tags within that block will be invalidated.
- If tag is an array tag or string tag, just this tag is invalidated.

3. If tag is to be written in the process of connecting:

• Write operation for the given tag will not take place.

Possible Causes/Solutions: "Frame contains errors"

1. Incorrect TPDU response size.

2. Unexpected frame received.

- Incorrect response code.

3. Frame sequence is out of order.

4. Device CPU work load is too high.

Solution:

1. Cable noise may cause distortion in the frame, resulting in erroneous data. It may also cause for dropped frames. Verify the cabling between the PC and the PLC device.

2. Reduce network traffic.

3. If this error occurs frequently, increase the Request Timeout and/or Fail After attempt count.

4. If this error occurs frequently, decrease the tag group scan rate to reduce the work load on the PLC CPU.

5. Increase the "Scan Cycle Load from Communication" and "Scan Cycle Monitoring Time".

"Device returned transport error [Code=<code>]"

An RFC1006 error (ISO over TCP/IP) occurred. This is the portion of the packet that encapsulates the S7 Messaging packet.

Solution:

Contact Technical Support.

"Device returned protocol error [Class=<class>, Code=<code>]"

An S7 Messaging error occurred. This will occur if this portion is malformed or contains incorrect packet lengths.

Solution:

Contact Technical Support.

Note:

No data access errors can occur for this operation.

See Also: Error Codes

Unable to read <block size> bytes starting at address <address> on device '<device name>'

Error Type:

Warning

Result:

1. If device error (i.e. transport, protocol or access) occurred.

- If tag is a block tag, the entire block (address -> address + block size) will be deactivated. All tags within that block will be deactivated. This is signified by the postfix message "...Block Deactivated."
- If tag is an array tag or string tag, just this tag is deactivated. This is signified by the postfix message "...Tag Deactivated."

2. If no device errors occurred.

- If tag is a block tag, the entire block will be invalidated. All tags within that block will be invalidated.
- If tag is an array tag or string tag, just this tag is invalidated.

Possible Causes/Solutions:

"Frame contains errors"

1. Incorrect TPDU response size.

2. Unexpected frame received.

- Incorrect response code.

3. Frame sequence is out of order.

4. Device CPU work load is too high.

Solution:

1. Cable noise may cause distortion in the frame, resulting in erroneous data. It may also cause for dropped frames. Verify the cabling between the PC and the PLC device.

- 2. Reduce network traffic.
- 3. If this error occurs frequently, increase the Request Timeout and/or Fail After attempt count.
- 4. Also if this error occurs frequently, decrease the tag group scan rate to reduce the work load on the PLC CPU.
- 5. Increase the "Scan Cycle Load from Communication" and "Scan Cycle Monitoring Time".

"NetLink returned error [Code=<code>]"

An error was returned from the PLC or NetLink adapter.

Solution:

1. If error code=0x11, users may have an incorrect MPI ID set. Determine the ID of the MPI through which communications is occurring and re-enter it in the MPI ID Device Property field.

2. If error code=0x87, users may be accessing data out of range in the device. Verify the address limits of the device and make corrections to the tag references causing the error.

3. For all other errors, contact Technical Support.

"Device returned transport error [Code=<code>]"

An RFC1006 error (ISO over TCP/IP) occurred. This is the portion of the packet that encapsulates the S7 Messaging packet.

Solution:

Contact Technical Support.

"Device returned protocol error [Class=<class>, Code=<code>]"

An S7 Messaging error occurred. This will occur if this portion is malformed or contains incorrect packet lengths.

Solution:

Contact Technical Support.

"Device returned data access error [Code=<code>]"

A data access error occurred. This will occur if, for instance, a requested address is out of range or is being referenced incorrectly.

Solution:

Contact Technical Support.

See Also: Error Codes

Unable to write to '<address>' on device '<device name>'

Error Type: Warning

Result: The write operation to address <address > failed.

Possible Causes/Solutions:

"Device not responding"

- 1. The connection between the device and the Host PC is broken.
- 2. The named device may have been assigned an incorrect IP address.
- 3. Device CPU work load is too high.

Solution:

- 1. Verify the cabling between the PC and the PLC device.
- 2. Verify the IP address given to the named device matches that of the actual device.

- 3. If this error occurs frequently, decrease the tag group scan rate to reduce the work load on the PLC CPU.
- 4. Increase the "Scan Cycle Load from Communication" and "Scan Cycle Monitoring Time".

"Frame contains errors"

1. Incorrect TPDU response size.

2. Unexpected frame received.

- Incorrect response code.
- 3. Frame sequence is out of order.

4. Device CPU work load is too high.

Solution:

1. Cable noise may cause distortion in the frame, resulting in erroneous data. It may also cause for dropped frames. Verify the cabling between the PC and the PLC device.

- 2. Reduce network traffic.
- 3. If this error occurs frequently, increase the Request Timeout and/or Fail After attempt count.
- 4. If this error occurs frequently, decrease the tag group scan rate to reduce the work load on the PLC CPU.
- 5. Increase the "Scan Cycle Load from Communication" and "Scan Cycle Monitoring Time".

"NetLink returned error [Code=<code>]"

An error was returned from the PLC or NetLink adapter.

Solution:

1. If error code=0x11, users may have an incorrect MPI ID set. Determine the ID of the MPI through which communications is occurring and re-enter it in the MPI ID Device Property field.

2. If error code=0x87, users may be accessing data out of range in the device. Verify the address limits of the device and make corrections to the tag references causing the error.

3. For all other errors, contact Technical Support.

"Device returned transport error [Code=<code>]"

An RFC1006 error (ISO over TCP/IP) occurred. This is the portion of the packet that encapsulates the S7 Messaging packet.

Solution:

Contact Technical Support.

"Device returned protocol error [Class=<class>, Code=<code>]"

An S7 Messaging error occurred. This will occur if this portion is malformed or contains incorrect packet lengths.

Solution:

Contact Technical Support.

"Device returned data access error [Code=<code>]"

A data access error occurred. This will occur if for instance, a requested address is out of range or is being referenced incorrectly.

Solution:

Contact Technical Support.

See Also: Error Codes

www.kepware.com

Index

Α

Address ' <address>' is out of range for the specified device or register</address>	63
Address Descriptions	
Address Validation	63
Addressing Options	7
Applicom Direct-Link SW1000 Item Syntax	46
Array size is out of range for address ' <address>'</address>	64
Array support is not available for the specified address:' <address>'</address>	64

В

BCD	. 38
Big Endian	7
Boolean	. 38

С

Cable Diagrams		35
Configuring Connections	5, 1	13
СР		5

D

Data Type ' <type>' is not valid for device address '<address>'</address></type>	64
Data Types Description	. 38
Device ' <device name="">' is not responding</device>	. 65
Device address ' <address>' contains a syntax error</address>	. 63
Device address ' <address>' is Read Only</address>	. 64
Device ID	. 5
Device Status Messages	. 65
Driver Error Messages	. 64
DWord	. 38

Ε

Error Codes	. 62
Error Descriptions	. 62

F

Float 3	38
---------	----

G

General Communications Parameters	. 5
-----------------------------------	-----

Η

Help Contents	. 4
How to Configure S7-1200 Connections with the Totally Integrated Automation (TIA) Portal.	29
How To Configure S7-200 Connections in Micro/WIN	. 8
How To Configure S7-300/400 Connections in STEP 7	25

I

INAT OPC Server TCPIPH1 Item Syntax	48
Internal Tags	42

L

LBCD	38
Legacy S7-300/400 Item Syntax	56
Little Endian	. 7
Long	38

Μ

MicroWin	. 8
Missing address	63

Ν

NetLink	. 5
NetLink: S7-300 Address Descriptions	41
NetLink: S7-400 Address Descriptions	42
Network	. 5

0

Optimizing Siemens TCP/IP Ethernet Communications	37
Overview	4

S

S7-1200 Address Descriptions	41
S7-200 Address Descriptions	39
S7-300 Address Descriptions	41
S7-400 Address Descriptions	41
S7 Communications Parameters	5
Short	38
Siemens Simatic Net Item Syntax	50
Siemens STEP 7 Item Syntax	52
Softing S7/S5 OPC Server Item Syntax	54
Standard S7-300/400/1200 Item Syntax	42
STEP 7	25

Т
U

Unable to connect to device <device name=""></device>	65
Unable to establish association with device <device name=""></device>	66
Unable to read <block size=""> bytes starting at address <address> on device <device name="">.</device></address></block>	67
Unable to write tag ' <address>' on device '<device name="">'</device></address>	68

W

Winsock initialization failed (OS Error = n) Winsock V1.1 or higher must be installed to use the Siemens TCP/IP Ethernet device	64
	65
driver	
Word	. 38