InTouch® Compact Edition

MOTCP Configuration Guide

Invensys Systems, Inc.

Revision A

Last Revision: 1/15/08

Copyright

© 2008 Invensys Systems, Inc. All Rights Reserved.

All rights reserved. No part of this documentation shall be reproduced, stored in a retrieval system, or transmitted by any means, electronic, mechanical, photocopying, recording, or otherwise, without the prior written permission of Invensys Systems, Inc. No copyright or patent liability is assumed with respect to the use of the information contained herein. Although every precaution has been taken in the preparation of this documentation, the publisher and the author assume no responsibility for errors or omissions. Neither is any liability assumed for damages resulting from the use of the information contained herein.

The information in this documentation is subject to change without notice and does not represent a commitment on the part of Invensys Systems, Inc. The software described in this documentation is furnished under a license or nondisclosure agreement. This software may be used or copied only in accordance with the terms of these agreements.

Invensys Systems, Inc.
26561 Rancho Parkway South
Lake Forest, CA 92630 U.S.A.
(949) 727-3200

http://www.wonderware.com

For comments or suggestions about the product documentation, send an e-mail message to productdocs@wonderware.com.

Trademarks

All terms mentioned in this documentation that are known to be trademarks or service marks have been appropriately capitalized. Invensys Systems, Inc. cannot attest to the accuracy of this information. Use of a term in this documentation should not be regarded as affecting the validity of any trademark or service mark.

Alarm Logger, ActiveFactory, ArchestrA, Avantis, DBDump, DBLoad, DT Analyst, Factelligence, FactoryFocus, FactoryOffice, FactorySuite, FactorySuite A^2, InBatch, InControl, IndustrialRAD, IndustrialSQL Server, InTouch, MaintenanceSuite, MuniSuite, QI Analyst, SCADAlarm, SCADASuite, SuiteLink, SuiteVoyager, WindowMaker, WindowViewer, Wonderware, Wonderware Factelligence, and Wonderware Logger are trademarks of Invensys plc, its subsidiaries and affiliates. All other brands may be trademarks of their respective owners.

Contents

Welcome **5**
Documentation Conventions 5
Technical Support 6

Chapter 1 About the MOTCP Driver **7**
About MOTCP Driver 7
Device Specifications 7
Network Specifications 8

Chapter 2 Configuring the MOTCP Driver **9**
Selecting the PLC Driver 10
Configuring PLC-specific Settings 11
Configuring Advanced Settings 13
Configuring Time-out Settings 13
Configuring Protocol Settings 14
Configuring Hand Shake Settings 15
Configuring Buffer Length Settings 16
Disabling and Enabling DTR and IR Settings 17
Associating a PLC Driver with an Access Name 18

Chapter 3 Troubleshooting the MOTCP Driver 21

Appendix A MOTCP Item References.....27

Data Types and Address Ranges 28
Device Registers 30
Other Type Settings 31
Register-Number Item Names.....33
TSX Quantum and TSX Momentum 33
Generic Modbus (4, 5 and 6-digit) 34
Item Names Using the Modicon PLC Register Addresses 34
Absolute Notation Item Names 36
Modulo-10000 Point Item Names.....37
Controller Function Codes 37

Index 39

Welcome

This guide explains how to configure the MOTCP driver for InTouch Compact Edition.

This guide assumes you know how to use Microsoft Windows, including navigating menus, moving between applications and moving objects on the screen. If you need help with these tasks, see the Microsoft online help.

You can view this document online or you can print it, in part or entirely, by using the print feature in Adobe Acrobat Reader.

Documentation Conventions

This documentation uses the following conventions:

Convention	Used for
Initial Capitals	Paths and file names.
Bold	Menus, commands, dialog box names, and dialog box options.
`Monospace`	Code samples and display text.

Technical Support

Wonderware Technical Support offers a variety of support options to answer any questions on Wonderware products and their implementation.

Before you contact Technical Support, refer to the relevant section(s) in this documentation for a possible solution to the problem. If you need to contact Technical Support for help, have the following information ready:

- The type and version of the operating system you are using.
- Details of how to recreate the problem.
- The exact wording of the error messages you saw.
- Any relevant output listing from the Log Viewer or any other diagnostic applications.
- Details of what you did to try to solve the problem(s) and your results.
- If known, the Wonderware Technical Support case number assigned to your problem, if this is an ongoing problem.

Chapter 1

About the MOTCP Driver

This section explains the concepts of the MOTCP driver and identifies all hardware and software components required to implement communication between InTouch Compact Edition and compatible PLCs.

About MOTCP Driver

The MOTCP driver enables communication between an InTouch Compact Edition application and devices using the Modbus protocol over TCP/IP.

Device Specifications

To establish communication with the MOTCP driver, your target PLC must meet the following specifications:

- Manufacturer: Modicon (or any device using the Modbus protocol communicating over TCP/IP)
- Compatible equipment:
  - AEG CPU 984 series
  - Any device that is fully compatible with the Modbus protocol
- Modicon programmer software: ModSoft
- Device run-time software: None

Network Specifications

To communicate with the target device, the network must meet the following specifications:

- Device communication port: Modbus ethernet port
- Physical protocol: ethernet/TCP-IP
- Logic protocol: Modbus
- Specific PC board: Any TCP/IP adapter (Ethernet board)

Chapter 2

Configuring the MOTCP Driver

This section explains how to configure the MOTCP Driver by completing the following tasks:

- Select the PLC driver
- Configure the connection type
- Configure other PLC-specific settings
- Configure advanced settings
- Associate an InTouch Access Name with the configured driver

You can configure the MOTCP driver from the **Driver Configuration** dialog box. In a managed InTouch compact application, you access this dialog box from the WindowMaker **Special**, **InTouch Compact Edition Application**, **IO Driver Configuration** menu.

For more information about configuring connection settings, see the documentation of your PLC hardware manufacturer.

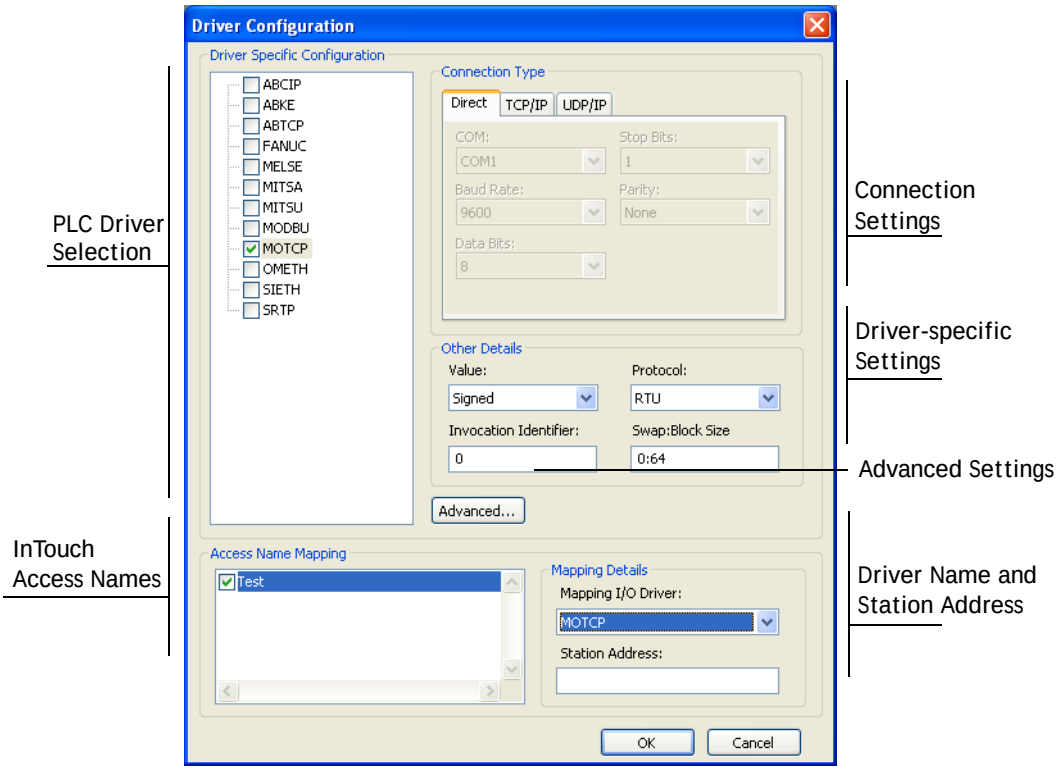
Note For each PLC type, you can configure only one set of Driver parameters.

Selecting the PLC Driver

Before you configure the PLC driver, you need to select it in the **Driver Configuration** dialog box. You can select more than one PLC driver to configure. All selected PLC drivers can be associated with InTouch Access Names.

To select the PLC driver

1. In WindowMaker click on **Special** on the menu bar, point to **InTouch Compact Edition Application** and then click **IO Driver Configuration**. The **Driver Configuration** dialog box appears.



2. From the list in the left pane, select the check box next to the PLC driver you want to configure.

   The selected PLC driver appears in the **Mapping I/O Driver** list in the **Mapping Details** area of the **Driver Configuration** dialog box. You can associate it with an InTouch Access Name. For more information, refer to Associating a PLC Driver with an Access Name on page 18.

You can now configure the connection settings, advanced settings, and other PLC-specific settings.

Configuring PLC-specific Settings

This section explains how to configure settings that are specific to the MOTCP Driver.

The following table lists the communication settings for the MOTCP driver and their corresponding values:

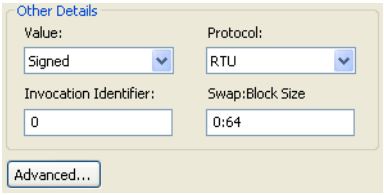
Parameters	Default Values	Valid Values	Description
Value	Signed	Signed or Unsigned	How integer values are handled by the device: • Signed - Allow negative values. Integer -32768 to 32767. Dword -2147483648 to 2147483647 • Unsigned - Does not allow negative values. Integer - 0 to 65535. DWord - 0 to 4294967295
Invocation identifier	0	0 to 32767	Used for transaction pairing when multiple words are sent along the same connection without waiting for a response
Protocol (ASCII or RTU)	RTU	RTU	Each eight-bit word is sent as two four-bit hexadecimal numbers, allowing for greater density and faster throughput. Note: RTU is the recommended protocol
		ASCII	Each eight-bit Word is sent as two four-bit ASCII characters, allowing for a time interval between characters without errors

Parameters	Default Values	Valid Values	Description
Swap:Block size	0:64	0 or 1 or 2 (Swap)	Option to change the order of data as it is processed • 0 - Word swap OFF; registers are not swapped • 1 - Word swap ON; swap words order for FP, FPS, FP3, FP3S, DW, DWS, DW3, DW3S, BCDDW, BCDDWS, BCDDW3 and BCDDW3S register types • 2 - Legacy swap; swap words order for FPS, FP3S, DWS, DW3S, BCDDWS and BCDDW3S register types only. The other register types are not swapped
		0 to 512 (Block Size)	The nominal size (in words) of each block of data to be transmitted, as determined by the processing capacity of the device

Note For more information about configuring MOTCP Driver-specific settings, see the documentation of your PLC hardware manufacturer.

To configure the MOTCP-specific settings

1. In the **Driver Configuration** dialog box, select the driver you want to configure.



2. In the **Other Details** area, assign values to the following settings:
   - **Value**
   - **Protocol**
   - **Invocation Identifier**
   - **Swap:Block Size**

Configuring Advanced Settings

This section explains how to configure advanced settings for a MOTCP driver. You can configure the following settings:

- Time-out settings
- Protocol settings
- Hand Shake settings
- Buffer Length settings
- Setting DTR and IR

Depending on the PLC driver you select, one or more of these settings might be disabled.

Note For more information about configuring advanced settings, see the documentation of your PLC hardware manufacturer.

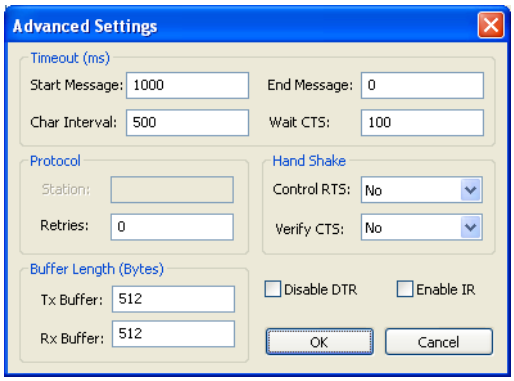
Configuring Time-out Settings

You can configure the following time-out settings:

- Start Message
- End Message
- Interval between characters
- Clear to Send wait

To configure time-out settings

1. In the **Driver Specific Configuration** dialog box, select the driver you want to configure.
2. Click **Advanced**. The **Advanced Settings** dialog box appears.



3. In the **Timeout** area, assign values to the following settings:

- **Start Message** - Time-out period in milliseconds for the message to start.
- **End Message** - Time-out period in milliseconds for the message to end.
- **Char Interval** - Time-out period in milliseconds between each character.
- **Wait CTS** - Time-out period in milliseconds for the Clear to Send wait.

4 Click **OK**.

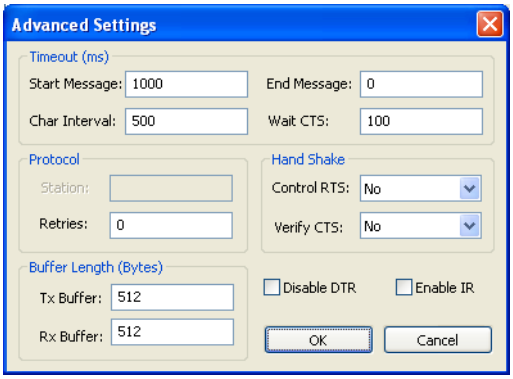
Configuring Protocol Settings

You can configure the number of attempts to retry a communication command after a previous failure. This setting specifies how many times the driver attempts to execute the same communication command before reporting a communication error.

To configure Protocol retry setting

1 In the **Driver Specific Configuration** dialog box, select the driver you want to configure.

2 Click **Advanced**. The **Advanced Settings** dialog box appears.



3 In the **Protocol** area, enter the number of retry attempts in the **Retries** box.

4 Click **OK**.

Configuring Hand Shake Settings

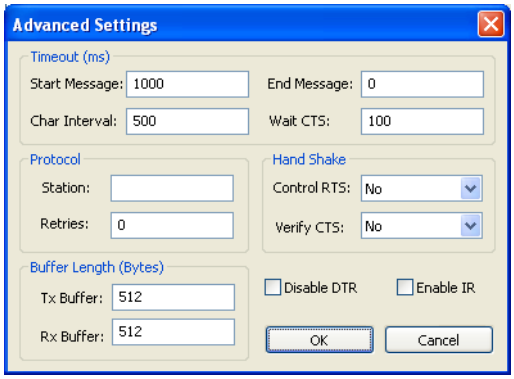
You can configure the following Hand Shake settings:

- Request to Send control
- Clear to Send verification

Hand Shake settings determine how the PLC driver communicates with the PLC over a communication channel after the physical establishment of the channel and before the normal information transfer.

To configure Hand Shake settings

1. In the **Driver Specific Configuration** dialog box, select the driver you want to configure.
2. Click **Advanced**. The **Advanced Settings** dialog box appears.



3. In the **Hand Shake** area, assign values to the following settings:
   - **Control RTS** - Specify whether to use the Request to Send control.The following options are available:

     No: Do not set the RTS (Request to Send) Hand Shake signal. Important: If you are using Windows 95/98 or Windows CE with the correct RS232/RS485 adapter (i.e. without RTS control), then you must select this option.

     Yes: Set the RTS (Request to Send) Hand Shake signal before communication. Important: If you are using Windows NT and the Cutler-Hammer RS232/RS485 adapter, then you must select this option.

     Yes+echo: Set the RTS (Request to Send) Hand Shake signal before communication, and echo the signal received from the target device.

     The default value is No.

- **Verify CTS** - Specify whether to use the Clear to Send type of verification.

  The default value is No.

4 Click **OK**.

Configuring Buffer Length Settings

You can configure the following Buffer Length settings:

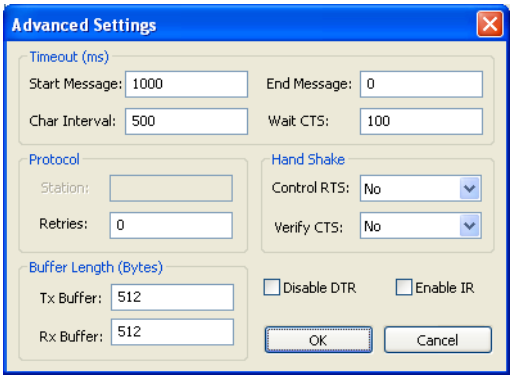
- Transmission Buffer (Tx) Length
- Reception Buffer (Rx) Length

The Buffer Length settings determine the number of bytes allocated from memory as a temporary buffer. The driver stores data in the temporary buffer while communicating with the PLC.

To configure Buffer Length settings

1 In the **Driver Specific Configuration** dialog box, select the driver you want to configure.

2 Click **Advanced**. The **Advanced Settings** dialog box appears.



3 In the **Buffer Length (Bytes)** area, assign values to the following settings:

- **Tx Buffer** - Transmission buffer size in bytes.
- **Rx Buffer** - Reception buffer size in bytes.

4 Click **OK**.

Disabling and Enabling DTR and IR Settings

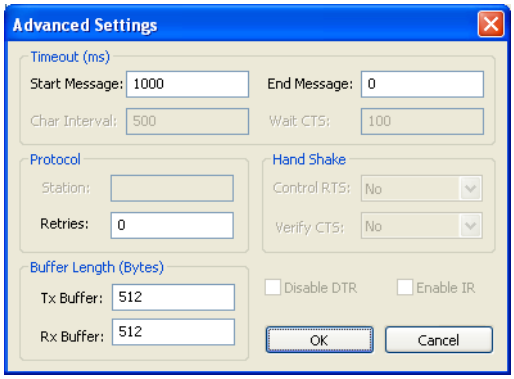
You can enable or disable the DTR and IR settings.

Setting DTR (Data Terminal Ready) means that the panel sends an additional signal to the PLC to indicate that the panel is ready to receive data.

Setting IR means that an infrared interface port is used instead of a standard serial port to communicate with the device.

To enable or disable DTR and IR settings

1. In the **Driver Specific Configuration** dialog box, select the driver you want to configure.
2. Click **Advanced**. The **Advanced Settings** dialog box appears.



3. Assign values to the following settings:
   - Select or clear the **Disable DTR** check box.
   - Select or clear the **Enable IR** check box.
4. Click **OK**.

Associating a PLC Driver with an Access Name

After configuring the PLC driver, you need to associate it with an existing InTouch Access Name. For more information about creating an InTouch Access Name, see the *InTouch HMI Data Management Guide*.

When you associate a PLC driver with an Access Name, you must also specify a station address. The station address is used by the MOTCP driver to specify slave network addresses of PLCs.

The station address in the Access Name should comply with the following syntax:

[*PLC Model*]:<*IP Address*>:[*IP Port Number*]:<*Slave ID*>

If the PLC model is not specified, the default is Generic 6.

PLC Model	MBTCP Object (see MBTCP Documentation)
TSX Quantum	TSX Quantum
TSX Momentum	TSX Momentum
TSX Premium	TSX Premium
Generic 4	Modbus PLC with register size=4
Generic 5	Modbus PLC with register size=5
Generic 6	Modbus PLC with register size=6
G4	Same as Generic 4
G5	Same as Generic 5
G6	Same as Generic 6

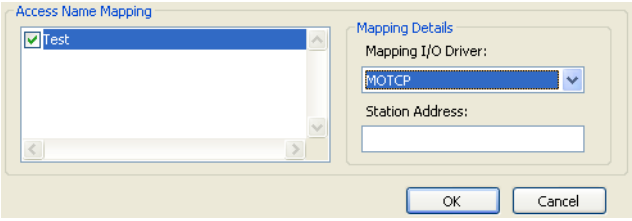
The Modbus PLC with register size = 4 means that the register addressing consists of four characters such as 4001. Likewise, G5 means that the register addressing consists of five characters such as 40001.

Note For more information about configuring the station address, see the documentation of your PLC hardware manufacturer.

To associate a PLC driver with an Access Name

1 Open the **Driver Configuration** dialog box.

2 In the **Access Name Mapping** area, locate the InTouch Access Name you want to associate.



Note If no Access Names appear in the list, you need to define an Access Name from WindowMaker.

3 Select the Access Name you want to associate.

4 In the **Mapping I/O Driver** list, select the configured driver.

5 In the **Station Address** box, type the station address.

When you move the mouse pointer over the **Station Address** box, a tooltip shows the required syntax for a station address.

6 Click **OK**.

Chapter 3

Troubleshooting the MOTCP Driver

You can detect communication errors between an InTouch Compact Edition application and the PLC by using the following item names:

Item name	Purpose
SysReadStatus	Receives error code for a failed read operation from the PLC
SysWriteStatus	Receives error code for a failed write operation to the PLC
SysReadCompleted	Indicates if the read operation from the PLC completes (TRUE) or not (FALSE)
SysWriteCompleted	Indicates if the write operation to the PLC completes (TRUE) or not (FALSE)

Note You can associate only one InTouch tag with each item name.

To troubleshoot communication problems

1. In WindowMaker, configure a set of I/O tags to associate with the error item names:
   - An I/O discrete tag to point at the item name SysReadCompleted.

- An I/O discrete tag to point at the item name SysWriteCompleted.
- An I/O integer tag to point at the item name SysReadStatus.
- An I/O integer tag to point at the item name SysWriteStatus.

2. Configure animation display links for the I/O tags, such as:
   - Fill color animations for the I/O tags associated with the SysReadCompleted and SysWriteCompleted items.
   - Analog value display links for the I/O tags associated with the SysReadStatus and SysWriteStatus items.
3. At run time, if a read or write error occurs between the InTouch Compact Edition and the PLC, the fill color animation indicates a problem. The value next to the fill color animation shows the error code.

Find the MOTCP error codes in the following table. The table also includes typical problems associated with an error code and possible solutions.

Error Code	Description	Possible Causes	Procedure to Solve
0	OK	Communication without problems	None required
1	Invalid serial port	The serial port specified is invalid	Specify a valid serial port
2	Invalid or unsupported item name	Item name specified is either invalid or unsupported	Check your item names for unsupported or invalid items
3	Illegal data value	Specified item name contains an invalid value that is not allowed for the slave	Check if the item name is supported by the device in use Verify if the tag value is valid
4	Illegal response strength	Equipment failed or is out of order, indicating that the framed request will generate a response whose size exceeds the available Modbus data size	Check the equipment state

Error Code	Description	Possible Causes	Procedure to Solve
5	Acknowledge	Error in the communication Ack action. Specialized use in conjunction with programming commands	Check the I/O driver configuration
6	Slave device busy	Invalid command if the equipment is in use. Specialized use in conjunction with programming commands	Studio commands never generate this error
7	Negative ack	Error in the communication ack action. Specialized use in conjunction with programming commands	Check the I/O Driver Configuration
8	Memory parity error	Incorrect communication parameter configuration. Specialized use in conjunction with function codes 20 and 21, to indicate that the extended file area failed to pass a consistency check	Check configuration of the driver's communication parameters
9	Out of memory	There is not enough memory for the I/O Server	Check the amount of RAM memory available for your system
10	Invalid or unsupported item name	Item name specified is either either invalid or unsupported	Check your item names for unsupported or invalid items
11	Invalid or unsupported item name	Item name specified is either either invalid or unsupported	Check your item names for unsupported or invalid items
12	Invalid block size	Internal error when generating the protocol block sizes	Contact Technical Support and send your published application
13	CheckSum error	Protocol error	Contact Technical Support and send your published application

Error Code	Description	Possible Causes	Procedure to Solve
14	Generic TCP or IP error	Wrong TCP / IP network configuration	Use Telnet or Ping tools to check your network configuration, and try to find the PLC with the computer
15	Invalid IP number	Wrong IP Address	You must specify the IP Address using the appropriate syntax—four fields, with up to three characters in each field, separated by periods
16	Connect error	Wrong IP Address or port number	Use Telnet or Ping tools to check your network configuration, and try to find the PLC with the computer
17	Time out between rx char	Response was not received from the device	Check the cable connection and also make sure that the I/O server settings match the PLC configuration
18	Invalid BCD value	Tried reading an invalid BCD value	Verify the PLC value is valid
19	Invalid BCD value	Tried writing a negative BCD value	Only positive BCD values are valid
32	Time out waiting for a tx message to finish	Started receiving a response but the communication was suspended	Check the cable connection and also make sure that the I/O Server settings match the PLC configuration
33	Invalid driver configuration file	Internal error	Contact Technical Support and send your published application
34	Invalid or unsupported item name	Item name specified is either invalid or unsupported	Check your item names for unsupported or invalid items
35	Driver API not initialized	Internal error	Contact Technical Support and send your published application
36	Invalid or unsupported item name	Item name specified is either invalid or unsupported	Check your item names for unsupported or invalid items

Error Code	Description	Possible Causes	Procedure to Solve
37	Invalid or unsupported item name	Item name specified is either invalid or unsupported	Check your item names for unsupported or invalid items
38	Invalid station	Station specified for this driver is invalid	Check the I/O driver configuration and specify a valid Station Address
39	Invalid block size	Internal error when generating the protocol block sizes	Contact Technical Support and send your published application
100	Invalid operation	Writing attempt in the input registers, Input Status, or Report Slave ID	You cannot write in these addresses
-15	Time out start message	Disconnected cables	Check cable wiring
		PLC is turned off, in stop mode, or in error mode	Check the PLC state – it must be RUN
		Wrong station number	Check the station number
		Wrong RTS/CTS control settings	Check the I/O driver configuration
-17	Time out between rx char	PLC in stop mode or in error mode	Check cable wiring
		Wrong station number	Check the PLC state – it must be RUN
		Wrong parity	Check the station number
		Wrong RTS/CTS configuration settings	Check the I/O driver configuration

Appendix

MOTCP Item References

This section provides MOTCP I/O driver reference information to configure the MOTCP driver.

If an “OUT OF RANGE” item is used in the Intouch Compact Edition application, then the quality for that item is shown as 0 or BAD. It is also possible that the quality of one or more items goes to 0 or BAD due to an “OUT OF RANGE” item being used.

Invalid values in items (with/without suffixes) can return unpredictable values. You should ensure that valid values are written for each data type. For example, 0 to 65535 is the valid value range for 'word' type of data.

The MOTCP driver supports float values with 4 bytes with up to 7 digits. Therefore, the valid range for float values is 3.4E +/- 38 (7 digits). Values outside this range result in an invalid value written to the tag. Furthermore, values with more than 7 digits of precision are rounded at the seventh digit.

The MOTCP driver supports valid BCD values from 0 to 9999 and valid LBCD values from 0 to 99999999. Invalid BCD values (e.g.: 1010 in binary base) result in an invalid value written to the tag.

Note The MOTCP driver does not support DBCS, only ANSI characters.

Data Types and Address Ranges

The following table lists the data types and address ranges that are valid for the MOTCP driver.

Type	Syntax	Valid Range	Comments
0X	0X:0	Varies by device	Coil status: Read and write events using Modbus instructions 01, 05, and 15
1X	1X:0	Varies by device	Input status: Read events using Modbus instruction 02
3X	3X:0	Varies by device	Input Register: Read events using Modbus instruction 04
4X	4X:0	Varies by device	Holding register: Read and write events using Modbus instructions 03, 06, and 16
FP	FP:0	Varies by device	Floating-point value (holding register): Read and write float-point values using two consecutive holding registers
FP3	FP3:0	Varies by device	Floating-point value (Input register): Read float-point values using two consecutive input registers
DF	DF:0	Varies by device	Double Precision Floating-point value (holding register): Read and Write double precision float-point values using four consecutive holding registers
DF3	DF3:0	Varies by device	Double Precision Floating-point value (Input register): Read double precision float-point values using four consecutive input registers
FP3S	FP3S:0	Varies by device	Floating-point value (Input register): Read float-point values using two consecutive input registers with Byte Swap
DW	DW:0	Varies by device	32-bit integer value (holding register): Read and write 32-bit integer values using two consecutive holding registers
FPS	FPS:0	Varies by device	Floating-point value (holding register): Read and write float-point values using two consecutive holding registers with Byte Swap

Type	Syntax	Valid Range	Comments
DWS	DWS:0	Varies by device	32-bit integer value (holding register): Read and write 32-bit integer values using two consecutive holding registers with Byte Swap
DW3	DW3:0	Varies by device	32-bit integer value (Input register): Read 32-bit integer values using two consecutive input registers
DW3S	DW3S:0	Varies by device	32-bit integer value (Input register): Read 32-bit integer values using two consecutive input registers with Byte Swap
BCD3	BCD3:0	Varies by device	BCD value (Input register): Read events using Modbus instruction 04
BCD	BCD:0	Varies by device	BCD Value (holding register): Read and write events using Modbus instructions 03, 06 and 16
BCDDW	BCDDW:0	Varies by device	BCD 32-bit integer value (holding register): Read and write 32-bit integer values using two consecutive holding registers
BCDDWS	BCDDWS:0	Varies by device	BCD 32-bit integer value (holding register): Read and write 32-bit integer values using two consecutive holding registers with byte Swap
BCDDW3	BCDDW3:0	Varies by device	BCD 32-bit integer value (Input register): Read 32-bit integer values using two consecutive input registers
BCDDW3S	BCDDW3S:0	Varies by device	BCD 32-bit integer value (Input register): Read 32-bit integer values using two consecutive input registers with byte swap
ID	ID:0	Varies by device	Report slave ID using Modbus instruction 17
ST	ST:0	Varies by device	String value (holding register): Reads and writes string values using consecutive holding registers
STS	STS:0	Varies by device	String value (holding register): Reads and writes string values using consecutive holding registers with byte swap

Device Registers

For examples of how device registers are specified using Header and Address, see the following table:

Device Register	Header	Address
00001	0x:1	0
00010	0x:0	10
01020	0x:1000	20
10001	1x:1	0
10010	1x:0	10
11020	1x:1000	20
30001	3x:1	30000
30010	3x:0	30010
31020	3x:1000	30020
30000 and 30001	FP3:30000	0
31001 and 31000	FP3S:0	31000
40001	4x:1	40001
40010	4x:40000	10
41020	4x:41000	20
40010 (bit 0)	4x:40000	10.0
41010 (bit 7)	4x:41000	10.7
00001 and 00002	FP:1	0
40013 and 40014	FP:40000	13
41021 and 41022	FP:41000	21
40000 to 40003	DF:30000	10000
1 to 4	DF3:1	0
41022 and 41021	FPS:1000	21
40001 and 40002	DW:1	0
40013 and 40014	DW:0	13
41021 and 41022	DW:1000	21
40002 and 40001	DWS:1	0
41021 and 41022	DW3:1000	21

Device Register	Header	Address
40011 and 40010	DW3S:1	10
40001 and 40002	BCD:1	0
41021 and 41022	BCD3:1000	21
40001 and 40002	BCDDW:1	0
41011 and 41010	BCDDWS:1000	10
40001 and 40002	BCDDW3:1	0
40011 and 40010	BCDDW3S:1	10
40001	ST:1:2	0
41010 and 41011	ST:1000:4	10
40001	STS:1:2	0
41011 and 41010	STS:1000:4	10

Other Type Settings

Parameter	In Touch CE Configuration
Network address	IP Address in the Station field
PLC unit ID	This parameter is the last field specified in the station
Reply time out (sec)	Specified in the MOTCP advanced settings (start time Out)
Maximum outstanding messages	Fixed in 1
Used concept data structures	This functionality can be expressed by using the SWAP option to 1 in the CE Driver configuration. Notice that this option can not be set differently for different devices as the Wonderware I/O server can. Also, you cannot have a different setting for reals and for longs
Bit order format	MOTCP supports only the B1 B2 … B16 order format
String variable style	Only C Style is supported (0 in the string will indicate end of string)

Parameter	In Touch CE Configuration
Maximum address ranges	CEView I/O Server does not impose limits. The PLC will return error if an invalid address is requested
Register type	Only binary is supported
Block I/O size	CEView divides the requests into blocks with at most 64 words (128 bytes)
Support multiple register write	Option not supported by CEView for the first release
Support multiple coil write	Option not supported by CEView for the first release
Register size	The user can specify the register size in the station field by using the models Generic 4, Generic 5, and Generic 6

Register-Number Item Names

TSX Quantum and TSX Momentum

Register Type	TSX Quantum/ TSX Momentum	Modicon Micro	Compact 984-265	Compact 984-145	Tag Type	Access	Supported by InTouch Compact Edition
Output coils	1-65536	1-9999	1-65536	1-9999	Discrete	Read/write	Yes
Contacts	100001-165536	10001-19999	100001-165536	10001-19999	Discrete	Read-only	Yes
Input	300001-365536	30001-39999	300001-365536	30001-39999	Analog	Read-only	Yes
Holding	400001-465536	40001-49999	400001-465536	40001-49999	Analog	Read/write	Yes
Extended	6x0000-6x9999			6x000-6x999	Analog	Read/write	No

TSX Quantum

Register Type	TSX Premium	Tag Type	Access	Supported by InTouch Compact Edition
Output coils	1-65536	Discrete	Read/write. Mapped to %M PLC object; that is, output coils and contacts registers are the same	Yes
Contacts	100001-165536	Discrete	Read-only. Mapped to %M PLC object; that is, output coils and contacts registers are the same	Yes
Input	300001-365536	Analog	Read-only. Mapped to %MW PLC object; that is, input registers and holding registers are the same	Yes
Holding	400001-465536	Analog	Read/write. Mapped to %MW PLC object; that is, input registers and holding registers are the same. Poking a value to a particular bit in the holding register in this TSX Premium PLC is not supported	Yes

Generic Modbus (4, 5 and 6-digit)

Register Type	Generic Modbus (6-Digit)	Generic Modbus (5-Digit)	Generic Modbus (4-Digit)	Tag Type	Access	Supported by InTouch Compact Edition
Output coils	1-65536	1-9999	1-999	Discrete	Read/write	Yes
Contacts	100001-165536	10001-19999	1001-1999	Discrete	Read-only	Yes
Input	300001-365536	30001-39999	3001-3999	Analog	Read-only	Yes
Holding	400001-465536	40001-49999	4001-4999	Analog	Read/write	Yes
Extended	6x0000-6x9999	6x000-6x999		Analog	Read/write	No

Item Names Using the Modicon PLC Register Addresses

Item Name	Description	Supported by InTouch Compact Edition
400001	When no spaces and no letters follow the register number, the register contents are treated as a 16-bit unsigned quantity	Yes
400001 S	When a space and the letter "S" follow the register number, the register contents are treated as a 16-bit signed quantity	Yes
400001 I	When a space and the letter "I" follow the register number, the register contents are treated as a 32-bit signed quantity This takes up two consecutive registers	Yes
400001 L	When a space and the letter "L" follow the register number, the register contents are treated as a 32-bit signed quantity	Yes
400001 F	When a space and the letter "F" follow the register number, the register contents are treated as a floating-point quantity. This takes up two consecutive registers	Yes

Item Name	Description	Supported by InTouch Compact Edition
400001 U	When a space and the letter "U" follow the register number, the register contents are treated as a 32-bit unsigned quantity	No
400001-400003 M	• When a space and the letter "M" follow the register number or register number pair separated by a dash, the register contents are treated as ASCII data • Each register contains up to two (2) ASCII characters • This example represents six (6) ASCII characters	Yes
300001:10	• When a colon and a number from 1 (one) to 16 follow the register number, the register contents are treated as discrete data • This example represents bit 10 of the input register 300001	Yes

Absolute Notation Item Names

Item Name	Description	Supported by InTouch Compact Edition
nnnnn **DO**	• Discrete output • Refers to the same data Modbus calls "coils" • Valid range is 0 (zero) **DO** through 65535 **DO**	Yes
nnnnn **DI**	• Discrete input • Refers to the same data called "contacts" by Modbus • Valid range is 0 (zero) **DI** through 65535 **DI**	Yes
nnnnn **IR**	• Input register • Refers to the same data called "input register" • Valid range is 0 (zero) **IR** through 65535 **IR**	Yes
nnnnn **HR**	• Holding register • Refers to the same data Modbus calls "holding register" • Valid range is 0 (zero) **HR** through 65535 **HR**	Yes
nnnnn **PV**	• Process variable • Refers to holding register, but treated as floating points and assumes two (2) registers per floating-point number • Valid range is 0 (zero) **PV** through 65535 **PV**	No

Modulo-10000 Point Item Names

Item Name	Description	Supported by InTouch Compact Edition
400001-400002	Can represent numbers between 0 and 99,999,999 Register 400001 = <9999> and register 400002 = <9999>	No
400005-400007	Can represent numbers between 0 and 2,147,483,646 Register 400005 = <21>, register 400006 = <4748>, and register 400007 = <3646>	No

Controller Function Codes

Code	Name	Description	Supported by InTouch Compact Edition
01 (0x01)	Read coils	Reads the ON/OFF status of discrete outputs (0X references, coils) in the slave	Yes
02 (0x02)	Read discrete inputs	Reads the ON/OFF status of discrete inputs (1XXXXX references) in the slave	Yes
03 (0x03)	Read holding registers	Reads the binary contents of holding registers (4XXXXX references) in the slave	Yes
04 (0x04)	Read input registers	Reads the binary contents of input registers (3XXXXX references) in the slave	Yes
05 (0x05)	Write single Coil	Forces a single coil (0X reference) to either ON or OFF	Yes
06 (0x06)	Write single register	Presets a value into a single holding register (4XXXXX reference)	Yes
15 (0x0F)	Write multiple coils	Forces each coil (0XXXXX reference) in a sequence of coils to either ON or OFF	No

Code	Name	Description	Supported by InTouch Compact Edition
16 (0x10)	Write multiple registers	Presets values into a sequence of holding registers (4XXXXX references)	Yes - only when writing 32 bit values
20 (0x14)	Read general reference	Returns the contents of registers in the extended memory file (6XXXXX) references	No
21 (0x15)	Write general reference	Writes the contents of registers in the extended memory file (6XXXXX) references	No
22 (0x16)	Mask write holding register	Modifies the contents of a specified (4XXXXX reference) holding register using a combination of an AND mask, an OR mask, and the register's current contents. The function can be used to set or clear individual bits in the register	No

Index

A

Access Name 18
advanced settings 13

B

Buffer Length Settings 16

D

documentation conventions 5
driver characteristics 8
DTR and IR settings 17

H

Hand Shake settings 15

I

item references
  address ranges 28
  controller function codes 37
  data types 28
  description 27
  device registers 30
  generic Modbus 34
  Modicon PLC register addresses 34
  TSX momentum 33
  TSX Quantum 33

M

MOTCP driver
  description 7
  device specifications 7
  jetwork specifications 8

P

PLC driver
  associating with an Access name 18
  PLC-specific settings 11
  selecting 10
protocol settings 14

T

technical support 6
Time-out Settings 13
troubleshooting
  communication problems 21
  configure animation display 22