

XS7PPI Driver Manual

Siemens S7-200 Advanced PPI Protocol Driver

SIEMENS

S7-200
PLC Series



CPKSoft Engineering

Process Monitoring and Industrial Automation Software

Copyright 1990-2008, CPKSoft Engineering. All rights reserved.

Index

1.	Introduction	3
2.	Driver details	4
2.1.	Driver overview	4
2.2.	Supported devices.....	4
3.	Command list	6
3.1.	Read Consecutive Variables as Analog Inputs (MB/MW/MD/VB/VW/VD/AI)	6
3.2.	Write Consecutive Variables as Analog Outputs (MB/MW/MD/VB/VW/VD/AQ)	7
3.3.	Read Consecutive Bits as Digital Inputs (I/Q).....	8
3.4.	Write Consecutive Bits as Digital Outputs (Q)	9
4.	Appendices	11
4.1.	Error messages	11
4.2.	Keywords list.....	12

1. Introduction

CPKSoft Engineering assumes no responsibility for any errors that may appear in this document. If you have any suggestions for improvements or amendments or have found errors in this publication, please notify us.

This driver is included with all unlimited licenses of TAS-HMITalk. It is not sold separately. It requires the TAS-HMITalk ActiveX to work, therefore it cannot be used as a stand-alone driver.

If you use this driver in your applications, you need to include the xs7ppi.tlk in the set of files that you distribute. This file must be located in the same folder where the hmitalk.ocx file is registered in order to be found by the activex when the applications are executed.

The source-code for the xs7ppi.tlk driver is available in plain-C language for additional USD 349 if you own a license of TAS-HMITalk 8.04 or higher.

Refer to the following link to visit the xs7ppi driver page at CPKSoft Engineering website: <http://www.cpksoft.com/tabid/55/ProductID/87/PageIndex/1/Default.aspx>.

Visit this link if you want to see a complete list of drivers that are currently available for TAS-HMITak: <http://www.cpksoft.com/Drivers/tabid/55/Default.aspx>.

Also, refer to this link if you are interested in purchasing a license of the most recent version of TAS-HMITalk: <http://www.cpksoft.com/Products/tabid/54/Default.aspx>.

We welcome your comments about this document. You can reach us by e-mail at [contact @ cpksoft.com](mailto:contact@cpksoft.com).

2. Driver details

2.1. Driver overview

The XS7PPI driver can be used to read or write data from or to PLCs of the S7-200 series that use the Siemens Point To Point Advanced PPI Communication Protocol over a RS-485 link multi-master networks using the Siemens multimaster cable configured as multimaster.

The RS-232 serial cable specification that was successfully tested with this driver is as follows:

- Cable PC/PPI multimaster
- 6ES7 901-3CB30-0XA0
- E-STAND:03

Suggested Cable Dip Switches Configuration:

- 1 0
- 2 1 9600 bauds
- 3 0
- 4 X Spare
- 5 1 Multimaster
- 6 0 Local DCE
- 7 0 11 bits
- 8 X Spare

Important note:

The suggested configuration will only work with TAS-HMITalk. If you want to communicate using MicroWin, you will need to set dip switch 5 to 0 in order to have the cable work as Freeport/PPI instead of Multimaster. Besides, you will need to uncheck the "Multimaster" and "Advanced PPI" options in the MicroWin configuration. Set dip switch back to 1 if you want to communicate using TAS-HMITalk.

COM SETTINGS:

Typical COM port settings are:

9600, N, 8, 1 9600, E, 8, 1 115200, N, 8, 1. 115200, E, 8, 1.

2.2. Supported devices

This driver can communicate with these devices, but is not necessarily limited to this list:

SIEMENS S7-200 PLC Series
SIEMENS S7-212 CPU
SIEMENS S7-214 CPU
SIEMENS S7-215 CPU
SIEMENS S7-216 CPU
SIEMENS S7-221 CPU
SIEMENS S7-222 CPU

SIEMENS S7-224 CPU
SIEMENS S7-277 EM

3. Command list

3.1. Read Consecutive Variables as Analog Inputs (MB/MW/MD/VB/VW/VD/AI)

Description of this command:

Reads a block of consecutive variables from a selected area, starting at a given offset.

Type of data handled by this command:

Analog Input

1-112 for MB and VB 1-56 for AI, MW and VW 1-28 for MD and VD

Meaning of the DriverP0 parameter:

0

Meaning of the DriverP1 parameter:

PC master address (0-127)

Meaning of the DriverP2 parameter:

Device destination address (0-127)

Meaning of the DriverP3 parameter:

Number of additional messages to gain access to the network (0-25).

Meaning of the DriverP4 parameter:

Stamp an automatic PDU_REF in each message (0=No, 1=Yes) (PDU_REF is a connection identification number that will be unique for each communication. Not necessary in pure master-slave networks).

Meaning of the DriverP5 parameter:

Indicates how to treat the data returned from the PLC:

0 = Treat data as unsigned

1 = Treat data as signed

Meaning of the DriverP6 parameter:

First variable in the block to be read. Variables that can be read are:

- AIxxxx
- MBxxxx
- MWxxxx
- MDxxxx
- VBxxxx
- VWxxxx
- VDxxxx For example: VB2000, VW1000, VD300, etc.

Meaning of the DriverP7 parameter:

Number of internal retries if communication timeouts (0-20). The driver will automatically make these number of additional attempts to communicate before returning a timeout error to the application. If left empty, 3 retries are assumed by default.

Values that are returned:

Value in PointValue (0) = First variable value

Value in PointValue (1) = Second variable value

...

Value in PointValue (HMITalk1.DriverNumPoints-1) = Last variable value

3.2. Write Consecutive Variables as Analog Outputs (MB/MW/MD/VB/VW/VD/AQ)

Description of this command:

Writes a block of consecutive variables to a selected area, starting at a given offset.

Important note: Make sure that the variables that you are trying

to write are not forced in the PLC. If the variables are forced, the PLC will successfully receive the data but it will not assume the new values. If you are writing a variable that could be forced, it is recommended that you read it back later to verify that the new value has been properly assumed by the PLC.

Type of data handled by this command:

Analog Output

1-112 for MB and VB 1-56 for AQ, MW and VW 1-28 for MD and VD

Meaning of the DriverP0 parameter:

0

Meaning of the DriverP1 parameter:

PC master address (0-127)

Meaning of the DriverP2 parameter:

Device destination address (0-127)

Meaning of the DriverP3 parameter:

Number of additional messages to gain access to the network (0-25).

Meaning of the DriverP4 parameter:

Stamp an automatic PDU_REF in each message (0=No, 1=Yes) (PDU_REF is a connection identification number that will be unique for each communication. Not necessary in pure master-slave networks).

Meaning of the DriverP5 parameter:

Indicates how to treat the data sent to the PLC:

0 = Treat data as unsigned

1 = Treat data as signed

Meaning of the DriverP6 parameter:

First variable in the block to be written. Variables that can be written are:

- AQxxxx
- MBxxxx
- MWxxxx
- MDxxxx
- VBxxxx
- VWxxxx
- VDxxxx For example: VB2000, VW1000, VD300, etc.

Meaning of the DriverP7 parameter:

Number of internal retries if communication timeouts (0-20). The driver will automatically make these number of additional attempts to communicate before returning a timeout error to the application. If left empty, 3 retries are assumed by default.

Important note:

If you receive an 'Invalid ROSCTR received' error message from the PLC, it could mean that the variable or variables do not exist in the PLC or that cannot be written.

Values that are sent:

Value in PointValue (0) = First variable value

Value in PointValue (1) = Second variable value

...

Value in PointValue (HMITalk1.DriverNumPoints-1) = Last variable value

3.3. Read Consecutive Bits as Digital Inputs (I/Q)

Description of this command:

Reads a block of consecutive bits from a selected area, starting at a given offset.

Type of data handled by this command:

Digital Input

Number of points accepted by this command:

1-896

Meaning of the DriverP0 parameter:

0

Meaning of the DriverP1 parameter:

PC master address (0-127)

Meaning of the DriverP2 parameter:

Device destination address (0-127)

Meaning of the DriverP3 parameter:

Number of additional messages to gain access to the network (0-25).

Meaning of the DriverP4 parameter:

Stamp an automatic PDU_REF in each message (0=No, 1=Yes) (PDU_REF is a connection identification number that will be unique for each communication. Not necessary in pure master-slave networks).

Meaning of the DriverP6 parameter:

First bit in the block of bits to be read. Variables that can be read are:

- Ixxx.b (with b=0 to 7)
- Qxxx.b (with b=0 to 7) For example: I0.0 or Q3.7.

Meaning of the DriverP7 parameter:

Number of internal retries if communication timeouts (0-20). The driver will automatically make these number of additional attempts to communicate before returning a timeout error to the application. If left empty, 3 retries are assumed by default.

Values that are returned:

Value in PointValue (0) = First bit value (0 or 1)

Value in PointValue (1) = Second bit value (0 or 1)

...

Value in PointValue (HMITalk1.DriverNumPoints-1) = Last bit value (0 or 1)

3.4. Write Consecutive Bits as Digital Outputs (Q)

Description of this command:

Writes a block of consecutive bits to a selected area, starting at a given offset.

Important note: Make sure that the bits that you are trying

to write are not forced in the PLC. If the bits are forced, the PLC will successfully receive the data but it will not assume the new values. If you are writing a bit that could be forced, it is recommended that you read it back later to verify that the new value has been properly assumed by the PLC.

Important note: It is recommended that you write in groups that are a

multiple of 8 bits and setting the starting bit to 0 (as in Q0.0, Q1.0, etc), to make sure that you are not unexpectedly changing other bits that belong to the same byte. If you want to change one single bit of a particular byte and you want to make sure that the other bits of the same byte will remain unchanged, it is up to your application to first read the complete byte in order to load all the unaffected bits with their proper values before the byte is written back to the PLC.

Type of data handled by this command:

Digital Output

Number of points accepted by this command:

1-896

Meaning of the DriverP0 parameter:

0

Meaning of the DriverP1 parameter:

PC master address (0-127)

Meaning of the DriverP2 parameter:

Device destination address (0-127)

Meaning of the DriverP3 parameter:

Number of additional messages to gain access to the network (0-25).

Meaning of the DriverP4 parameter:

Stamp an automatic PDU_REF in each message (0=No, 1=Yes) (PDU_REF is a connection identification number that will be unique for each communication. Not necessary in pure master-slave networks).

Meaning of the DriverP6 parameter:

First bit in the block of bits to be written. Bits that can be written are:

- Qxxx.x For example: Q3.7.

Meaning of the DriverP7 parameter:

Number of internal retries if communication timeouts (0-20). The driver will automatically make these number of additional attempts to communicate before returning a timeout error to the application. If left empty, 3 retries are assumed by default.

Important note:

If you receive an 'Invalid ROSCTR received' error message from the PLC, it could mean that the bit or bits do not exist in the PLC or that cannot be written.

Values that are sent:

Value in PointValue (0) = First bit value (0 or 1)

Value in PointValue (1) = Second bit value (0 or 1)

...

Value in PointValue (HMITalk1.DriverNumPoints-1) = Last bit value (0 or 1)

4. Appendices

4.1. Error messages

The following list shows all the possible error messages that can be returned by the protocol driver during a failed communication in the 'DriverStatus' property.

This list does not include some error messages that can be returned by the activex component while attempting to establish a connection.

- [1005] DRIVER (Internal): Invalid driver stage
- [1103] DRIVER (Port): Error reading current port status
- [1104] DRIVER (Port): Error writing port settings
- [1300] PROTOCOL (Timeout): No answer
- [1310] PROTOCOL (Timeout): No answer within expected time
- [1312] PROTOCOL (Timeout): Couldn't get attention from PLC
- [1410] PROTOCOL (Format): Invalid device id in response
- [1412] PROTOCOL (Format): Invalid number of bytes received
- [1433] PROTOCOL (Format): Validation error in device response
- [1436] PROTOCOL (Format): Invalid service_id received
- [1437] PROTOCOL (Format): Invalid ROSCTR received
- [2284] CONFIG (NumValues): Too many values (max=112)
- [2285] CONFIG (NumValues): Too many values (max=56)
- [2286] CONFIG (NumValues): Too many values (max=28)
- [3001] CONFIG (P0): Invalid command
- [3589] CONFIG (P1): Invalid destination address (0-127)
- [4135] CONFIG (P2): Invalid source address (0-127)
- [4550] CONFIG (P3): Invalid number of retries (0-25)
- [5034] CONFIG (P4): Invalid setting (0 o 1)
- [5515] CONFIG (P5): Invalid setting (0 o 1)
- [6037] CONFIG (P6): Invalid variable
- [8367] CONFIG (Remote): Wrong number of data bytes
- [8379] CONFIG (Remote): Error in the application ID of the request
- [8380] CONFIG (Remote): Error in the object definition (e.g. bad data type)
- [8381] CONFIG (Remote): No resources available
- [8382] CONFIG (Remote): Error in the structure of the service request
- [8383] CONFIG (Remote): Error in the communication equipment
- [8384] CONFIG (Remote): Access error
- [8385] CONFIG (Remote): OVS error
- [8386] CONFIG (Remote): Diagnostic error
- [8387] CONFIG (Remote): Protection system error
- [8388] CONFIG (Remote): BuB error
- [8389] CONFIG (Remote): Layer 2 specific error
- [8390] CONFIG (Remote): Invalid PDU_REF in response
- [8391] CONFIG (Remote): Hardware fault
- [8392] CONFIG (Remote): Illegal object access

- [8393] CONFIG (Remote): Invalid address (incorrect variable address)
- [8394] CONFIG (Remote): Data type is not supported (currently, only octet string is supported)
- [8395] CONFIG (Remote): Object does not exist or length error
- [8396] CONFIG (Remote): Unknown ERR_CLS error
- [8397] CONFIG (Remote): Unknown access error

4.2. Keywords list

The following list shows a set of words directly related to this driver.

"Advanced, CPU, EM, PLC, PPI, S7-200, S7-212, S7-214, S7-215, S7-216, S7-221, S7-222, S7-224, S7-277, Series, SIEMENS".