Industrial communication
solutions for Windows

XS7MPITP Driver Manual

Siemens S7-300/400 MPI Protocol ISO over TCP Driver

Contents

CPKSoft Engineering
Industrial communication drivers.

www.cpksoft.com
www.facebook.com/
cpksoftengineering
cpksoftengineering@
hotmail.com
phone: 54-911-45788354

1990-2012

XS7MPITP technical specifications

General information

The XS7MPITP driver can be used to read or write data from or to PLCs of the S7-300/400 series with the MPI protocol in ISO over TCP mode.
This driver requires an ethernet connection. It cannot be used through a serial link.

Command list

Send Connect Request Command

Description of this command:
Sends a Connect Request command to the PLC ethernet link. This must be the first command issued before any other command, to gain the attention of the PLC.
Methods used to run this command:
Analog Output
Number of points accepted by this command:
1
Meaning of the DriverP0 parameter:
100
Meaning of the DriverP1 parameter:
TPDU Destination reference (typically 0)
Meaning of the DriverP2 parameter:
TPDU Source reference (typically 1)
Meaning of the DriverP3 parameter:
Source-TSAP (typically 0100h)
Meaning of the DriverP4 parameter:
Destination-TSAP (typically 0102h)
Meaning of the DriverP5 parameter:
TPDU Size (8=256 bytes, 9=512 Bytes, 10=1024 bytes<--recomended)
Values that are sent:
Value in PointValue (0) = Ignored

Send Open S7 Connection Command

Description of this command:
Sends an Open S7 Connection command to the PLC. This must be the second command issued to the PLC, after the Connect Request command and before any read or write command. It starts a conversation with the PLC, setting a start sequence number that will be incremented for the remaining commands.
Methods used to run this command:
Analog Output
Number of points accepted by this command:
1
Meaning of the DriverP0 parameter:
101
Meaning of the DriverP1 parameter:
Start value for PDU_REF sequence number (typically 2)
Meaning of the DriverP4 parameter:
Sequence number mode (PDU_REF), where:
- 0 = Use the value indicated in DriverP1 for this telegram
- 2 = Internally generate an unique value based in the PC timer

Meaning of the DriverP6 parameter:
Ignore validations in response:
- Empty = No
- 1 = Yes

Values that are sent:

Value in PointValue (0) = Ignored

Read Multiple Registers (BYTE/WORD/DWORD)

Description of this command:

Reads a set of consecutive registers from a selected area, starting at a given start address.

Methods used to run this command:

Analog Input

Number of points accepted by this command:

Number of values

- 1-112 for BYTE type
- 1-56 for WORD type
- 1-28 for DWORD type

Meaning of the DriverP0 parameter:

Type

- 2 = BYTE
- 4 = WORD
- 6 = DWORD
- Use 4 for analog inputs of 200 family
- Use 4 for analog outputs of 200 family
- Use 28 for S7 counters
- Use 29 for S7 timers
- Use 30 for IEC counters (200 family)
- Use 31 IEC timers (200 family)

Meaning of the DriverP1 parameter:

DB Number

Meaning of the DriverP2 parameter:

Area

- 3 = System info of 200 family
- 5 = System flags of 200 family
- 6 = Analog inputs of 200 family
- 7 = Analog outputs of 200 family
- 80h = Direct peripheral access
- 81h = Unknown
- 82h = Unknown
- 83h = Unknown
- 84h = Data blocks
- 85h = Instance data blocks
- 86h = Not tested
- 87h = Unknown
- 28 = S7 counters
- 29 = S7 timers
- 30 = IEC counters (200 family)
- 31 = IEC timers (200 family)
- 86 = System data area
- 0 = Raw memory

Meaning of the DriverP3 parameter:

Start address

Meaning of the DriverP4 parameter:

Sequence number mode (PDU_REF), where:

- 0 = Use always the value indicated in the Open S7 Connection Command
- 1 = Increment the value indicated in the Open S7 Connection Command
- 2 = Internally generate an unique value based in the PC timer

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:

Ignore validations in response:

- Empty = No
- 1 = Yes

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

Write Multiple Registers (BYTE/WORD/DWORD)

Description of this command:
Writes a set of consecutive registers to a selected area, at a given starting address.
Important note:Make sure that the registers 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 writting a variable that could be forced, it is recomended that you read it back later to verify that the new value has been properly assumed by the PLC.
Methods used to run this command:
Analog Output
Number of points accepted by this command:
Number of values
- 1-112 for BYTE type
- 1-56 for WORD type
- 1-28 for DWORD type

Meaning of the DriverP0 parameter:
Type
- 2 = BYTE
- 4 = WORD
- 6 = DWORD
- Use 4 for analog inputs of 200 family
- Use 4 for analog outputs of 200 family
- Use 28 for S7 counters
- Use 29 for S7 timers
- Use 30 for IEC counters (200 family)
- Use 31 IEC timers (200 family)

Meaning of the DriverP1 parameter:
DB Number
Meaning of the DriverP2 parameter:
Area
- 3   = System info of 200 family
- 5   = System flags of 200 family
- 6   = Analog inputs of 200 family
- 7   = Analog outputs of 200 family
- 80h = Direct peripheral access
- 81h = Unknown
- 82h = Unknown
- 83h = Unknown
- 84h = Data blocks
- 85h = Instance data blocks
- 86h = Not tested
- 87h = Unknown
- 28 = S7 counters
- 29 = S7 timers
- 30 = IEC counters (200 family)
- 31 = IEC timers (200 family)
- 86 = System data area
- 0   =   Raw memory

Meaning of the DriverP3 parameter:
Start address
Meaning of the DriverP4 parameter:
Sequence number mode (PDU_REF), where:
- 0 = Use always the value indicated in the Open S7 Connection Command
- 1 = Increment the value indicated in the Open S7 Connection Command
- 2 = Internally generate an unique value based in the PC timer

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:
Ignore validations in response:
- Empty = No
- 1 = Yes

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

Error messages

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

[1005] DRIVER (Internal): Invalid driver stage
[1300] PROTOCOL (Timeout): No answer
[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
[1442] PROTOCOL (Format): Invalid PDU header received
[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

Supported devices

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

SIEMENS S7-315
SIEMENS S7 CPU IM151-8 PN/DP
SIEMENS S7 ET200S
SIEMENS S7-300 Series with MMI/Ethernet port
SIEMENS S7-400 Series with MMI/Ethernet port

CPKSoft Engineering
Industrial communication drivers.
www.cpksoft.com
www.facebook.com/cpksoftengineering
cpksoftengineering@hotmail.com
phone: 54-911-45788354

1990-2012