

Manual | EN

TS6610

TwinCAT 2 | PLC S5/S7 Communication



Table of contents

1 Foreword	5
1.1 Notes on the documentation	5
1.2 For your safety	5
1.3 Notes on information security.....	7
2 Overview	8
3 Introduction	10
4 Description of the PLC data area parameter values	12
5 Function blocks	14
5.1 FB_NLinkOpen.....	14
5.2 FB_NLinkClose	15
5.3 FB_NLinkFetch	16
5.4 FB_NLinkWrite	17
6 Functions	19
6.1 F_GetVersionTcS5S7Comm.....	19
7 Data types	20
7.1 E_NLinkArea	20
7.2 E_NLinkProtocol	20
7.3 E_NLinkTransport	20
7.4 E_NLinkOpState	21
7.5 ST_NLinkDebug.....	21
7.6 ST_NLinkSettings	21
7.7 T_HNLINK.....	22
8 Constants	23
9 Error codes	24
9.1 Internal S5/S7 communication error codes	24
9.2 Troubleshooting/diagnostics	26
10 Samples	28

1 Foreword

1.1 Notes on the documentation

This description is intended exclusively for trained specialists in control and automation technology who are familiar with the applicable national standards.

For installation and commissioning of the components, it is absolutely necessary to observe the documentation and the following notes and explanations.

The qualified personnel is obliged to always use the currently valid documentation.

The responsible staff must ensure that the application or use of the products described satisfies all requirements for safety, including all the relevant laws, regulations, guidelines, and standards.

Disclaimer

The documentation has been prepared with care. The products described are, however, constantly under development.

We reserve the right to revise and change the documentation at any time and without notice.

No claims to modify products that have already been supplied may be made on the basis of the data, diagrams, and descriptions in this documentation.

Trademarks

Beckhoff®, TwinCAT®, TwinCAT/BSD®, TC/BSD®, EtherCAT®, EtherCAT G®, EtherCAT G10®, EtherCAT P®, Safety over EtherCAT®, TwinSAFE®, XFC®, XTS® and XPlanar® are registered and licensed trademarks of Beckhoff Automation GmbH.

If third parties make use of designations or trademarks used in this publication for their own purposes, this could infringe upon the rights of the owners of the said designations.

Patents

The EtherCAT Technology is covered, including but not limited to the following patent applications and patents:

EP1590927, EP1789857, EP1456722, EP2137893, DE102015105702
and similar applications and registrations in several other countries.

EtherCAT®

EtherCAT® is registered trademark and patented technology, licensed by Beckhoff Automation GmbH, Germany

Copyright

© Beckhoff Automation GmbH & Co. KG, Germany.

The distribution and reproduction of this document as well as the use and communication of its contents without express authorization are prohibited.

Offenders will be held liable for the payment of damages. All rights reserved in the event that a patent, utility model, or design are registered.

1.2 For your safety

Safety regulations

Read the following explanations for your safety.

Always observe and follow product-specific safety instructions, which you may find at the appropriate places in this document.

Exclusion of liability

All the components are supplied in particular hardware and software configurations which are appropriate for the application. Modifications to hardware or software configurations other than those described in the documentation are not permitted, and nullify the liability of Beckhoff Automation GmbH & Co. KG.

Personnel qualification

This description is only intended for trained specialists in control, automation, and drive technology who are familiar with the applicable national standards.

Signal words

The signal words used in the documentation are classified below. In order to prevent injury and damage to persons and property, read and follow the safety and warning notices.

Personal injury warnings**⚠ DANGER**

Hazard with high risk of death or serious injury.

⚠ WARNING

Hazard with medium risk of death or serious injury.

⚠ CAUTION

There is a low-risk hazard that could result in medium or minor injury.

Warning of damage to property or environment**NOTICE**

The environment, equipment, or data may be damaged.

Information on handling the product

This information includes, for example:
recommendations for action, assistance or further information on the product.

1.3 Notes on information security

The products of Beckhoff Automation GmbH & Co. KG (Beckhoff), insofar as they can be accessed online, are equipped with security functions that support the secure operation of plants, systems, machines and networks. Despite the security functions, the creation, implementation and constant updating of a holistic security concept for the operation are necessary to protect the respective plant, system, machine and networks against cyber threats. The products sold by Beckhoff are only part of the overall security concept. The customer is responsible for preventing unauthorized access by third parties to its equipment, systems, machines and networks. The latter should be connected to the corporate network or the Internet only if appropriate protective measures have been set up.

In addition, the recommendations from Beckhoff regarding appropriate protective measures should be observed. Further information regarding information security and industrial security can be found in our <https://www.beckhoff.com/secguide>.

Beckhoff products and solutions undergo continuous further development. This also applies to security functions. In light of this continuous further development, Beckhoff expressly recommends that the products are kept up to date at all times and that updates are installed for the products once they have been made available. Using outdated or unsupported product versions can increase the risk of cyber threats.

To stay informed about information security for Beckhoff products, subscribe to the RSS feed at <https://www.beckhoff.com/secinfo>.

2 Overview

The function blocks of the PLC library **TcS5S7Comm.Lib** can be used to realise data exchange between the local TwinCAT PLC and a S5/S7 controller via an Ethernet LAN adapter from IBHsoftec.

The adapter is not part of the Beckhoff product and must be bought from IBHsoftec.

System requirements:

Supported adapters:

- **IBH Link S5++** adapter from IBHsoftec (**from firmware version 1.39**; the firmware version can be read via the configuration software provided).
Older adapters (without ++) are currently not supported.
- **IBH Link S7++** adapter from IBHsoftec (from firmware version **NL50MPI V02.004 05.06.09**; the firmware version can be read via the configuration software provided).

Programming environment:

- Windows NT (XP, XPe/WES, Vista, W7, WES7, ...);
- TwinCAT installation level: TwinCAT PLC or higher;
- TwinCAT system version 2.11.0 build >= 1537 or higher;

Target system type:

- Industrial PC or Embedded PC/CX (x86, ARM);
- Operating system:
 - Windows NT (XP, XPe/WES, Vista, W7, WES7, ...);
 - Windows CE (x86) (image v3.08c or higher);
 - Windows CE (ARM) (image v2.22c or higher);
- TwinCAT PLC runtime system version 2.10.0 or higher;

Product components

- **TcS5S7Comm.Lib** (library with communication blocks to be integrated in your PLC application);
- Tcplp.Lib (implements basic TCP/IP and UDP functions);
- TcSocketHelper.Lib (implements auxiliary TCP/IP functions);
- TwinCAT TCP/IP connection server (TwinCAT server);

Installation

Windows NT (XP, XPe/WES, Vista, W7, WES7, ...)

The PLC libraries are copied into the ...\\TwinCAT\\PLC\\Lib folder. The TwinCAT TCP/IP connection server is entered into the list of TwinCAT servers. The TCP/IP Connection Server is automatically started when TwinCAT is started and stopped when TwinCAT is stopped.

Windows CE

The following steps are required if you have obtained a product version for Windows CE:

- First install the product as usual on your programming PC. The PLC libraries are copied into the ... \\TwinCAT\\PLC\\Lib folder.
- X86 CPU (CX1000, CX1020, ...):
 - After the installation you will find a cabinet file for the CE runtime system in folder ... \\TwinCAT\\CE\\TCPIP\\Install.
 - Copy the file **TcTCPIPSvrCe.I586.CAB** into a folder in the CE runtime system.
- ARM CPU (CX9000, ...):

- After the installation you will find a cabinet file for the CE runtime system in folder ...
\TwinCAT\CE\TCPIP\Install\.
- Copy the file **TcTCPIPSvrCe.ARMV4I.CAB** in a folder in the CE runtime system.
- In the CE system: Install the CE components (by double-clicking on the cabinet file).
- Reboot the CE device. The TwinCAT TCP/IP connection server is automatically started with the CE operating system.

3 Introduction

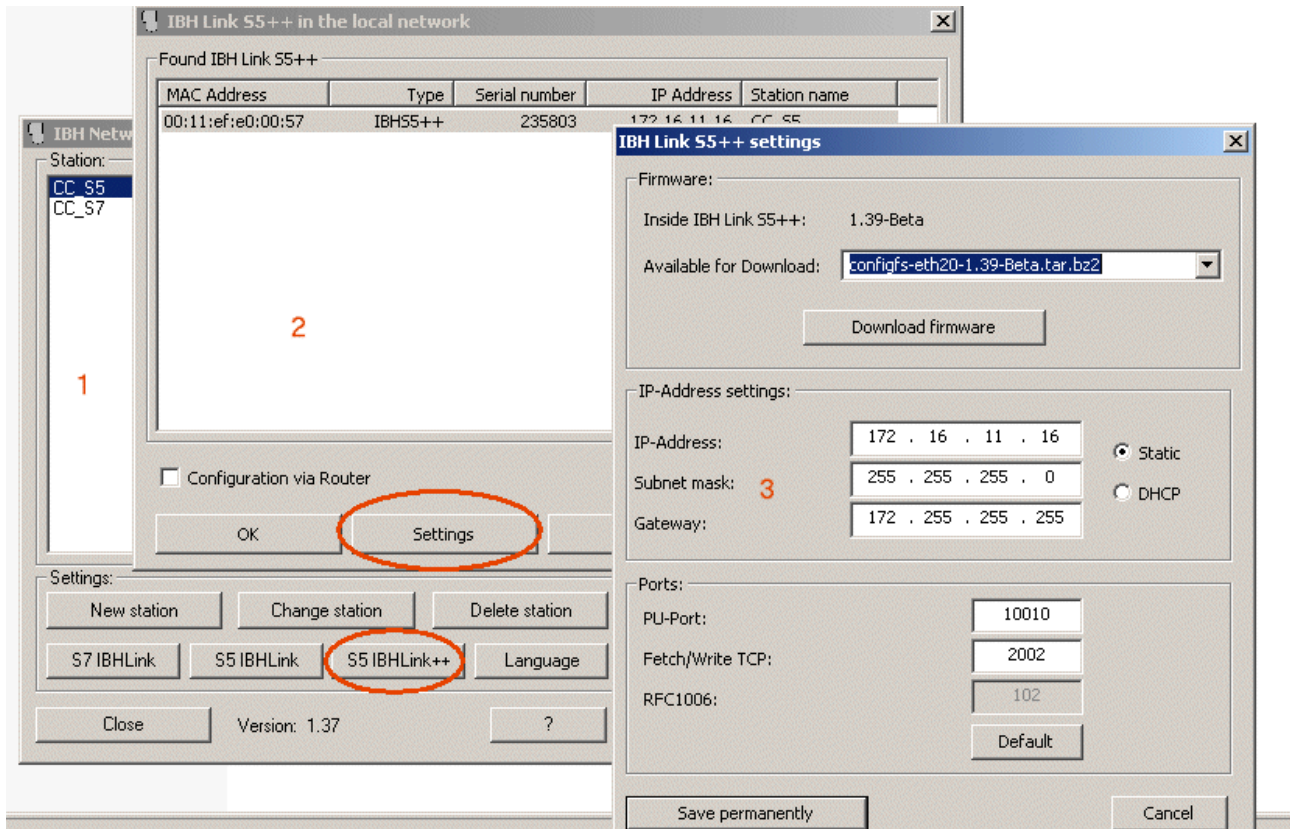
A detailed description of the module configuration can be found in the manufacturer documentation.

Installation notes can be found in the [overview page \[▶ 8\]](#).

The [examples \[▶ 28\]](#) page contains simple PLC project examples including all sources.

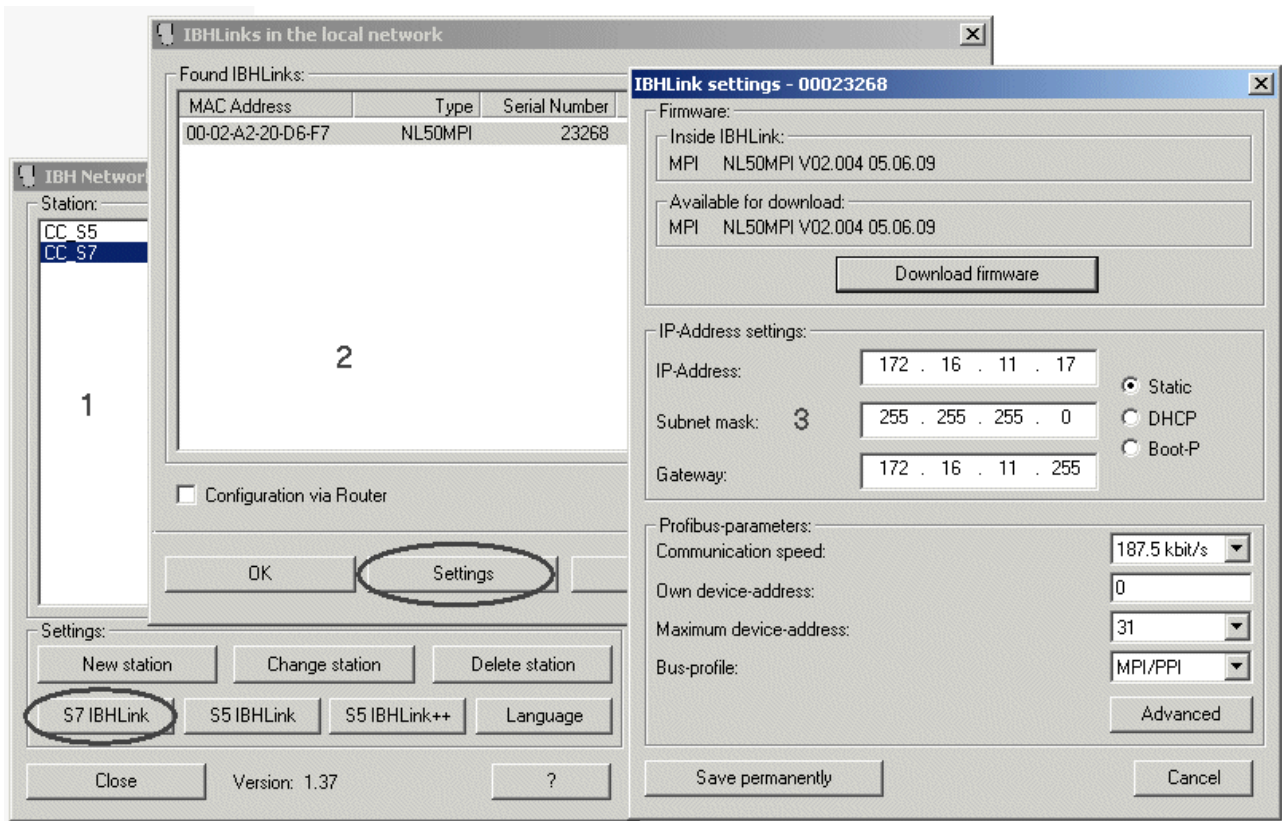
Configuration of an IBH Link S5++ adapter

In the following example a fixed IP address 172.16.11.16 was set:



Configuration of an IBH Link S7++ adapter

In the following example a fixed IP address 172.16.11.17 was set:



4 Description of the PLC data area parameter values

The parameter values to be used for the write/read command differ depending on the type of external control used, the adapter or the communication protocol. Some data areas may not be available, or the data have to be read/written word by word instead of byte by byte. Further information can be found in the documentation for the external control.

Format conversion necessary

Read access to word variables in the S5/S7 controller returns data in Motorola format (Hi and Lo bytes interchanged). For the evaluation the read data first have to be converted to the common Intel format used in the TwinCAT system. Data to be written from the TwinCAT system to the S5/S7 controller must be converted to the Motorola format before the write access. The BCD number system is used in the S5/S7 controller for representing the counter/timer variables. The data may have to be converted, if necessary.



The TwinCAT PLC library TcUtilities.Lib offers functions for conversion between Motorola and Intel format (e.g. BE16_TO_HOST, HOST_TO_BE16...).

The following table provides an overview of the available communication paths:

Control	LAN adapter	Transport protocol	LAN adapter communication protocol	Description
S5	IBH Link S5++	TCP/IP	IBH S5 LAN link protocol = FETCH/WRITE telegrams [▶ 12]	No connection parameterization required in the external control.
S7	IBH Link S7++	TCP/IP	NetLink MPI protocol (NL-MPI) from Hilscher [▶ 12]	No connection parameterization required in the external control.

IBH S5 LAN link protocol = FETCH/WRITE telegrams

- This protocol can currently only be used with S5 controllers;
- The TCP/IP port number has the default value: 2002;
- The parameter devAddr is not evaluated;
- The data block number (nDB) is only evaluated when the areas data blocks (DB) and extended data blocks (DX) are accessed. Permissible range: 1..255;
- The areas P (periphery) and OS (operating state) are not supported;

Data area (eArea)	Start address (nOffset)	Permitted data length (cbBuffer)
DB	Data word offset: 0..2047	read/write 1..2048 words
M	Flag byte offset: 0..255	read/write 1..256 bytes
I	Input byte offset: 0..127	read 1..128 bytes
Q	Output byte offset: 0..127	write 1..128 bytes
P	I/O peripheral byte offset: 0..127 (digital I/Os), 128..255 (analog I/Os)	read/write 1..256 bytes
C	Counter start offset: 0..255	read/write 1 word
T	Timer start offset: 0..255	read/write 1 word
DX	Data word offset	read/write 1..2048 words

NetLink MPI protocol (NL-MPI) from Hilscher

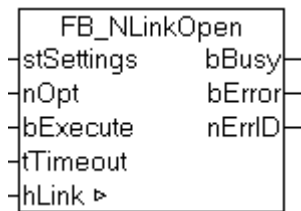
- This protocol can only be used with S7 controllers;
- The TCP/IP port number has the default value: 1099
- The MPI address (devAddr) has the default value: 2. Permissible values: 1..126;
- The data block number (nDB) is only evaluated when the data blocks (DB) data area is accessed. Permissible values: 0-65534;

- The areas P (periphery) and DX (extended data block) are not supported;
- The byte length has to be specified for the data length parameter (cbBuffer);
- The OS area (operating state) is only available for read access;

Data area (eArea)	Start address (nOffset)	Permitted data length (cbBuffer)
DB	Data byte offset: 0..65534	Read: max. 222 bytes, write: max. 216 bytes
M	Flag byte offset: 0..65534	
I	Input byte offset: 0..65534	
Q	Output byte offset: 0..65534	
C	Counter start offset: 0..65534	Read: max. 111 word counters, write: max. 108 counters
T	Timer start offset: 0..65534	Read: max. 111 word timers, write: max. 108 timers
OS	Operating state: irrelevant	1 word (0=STOP, 1=START, 2=RUN, 3=UNKNOWN)

5 Function blocks

5.1 FB_NLinkOpen



This function block (actively) establishes a network connection to the third-party system. If successful, the internal parameters of the *hLink* variable are initialised. The same instance of the *hLink* variable must then be transferred to the other instances of the communication blocks [FB_NLinkFetch](#) [▶ 16], [FB_NLinkWrite](#) [▶ 17] or [FB_NLinkClose](#) [▶ 15]. In a PLC application one instance of this variables is required for each network connection (TwinCAT system<->Ethernet LAN adapter<->external controller).

VAR_IN_OUT

```
VAR_IN_OUT
  hLink : T_HNLINK;
END_VAR
```

hLink : [Network connection handle](#) [▶ 22]. The instance of this variable is required by the other communication blocks and is transferred to them via VAR_IN_OUT.

VAR_INPUT

```
VAR_INPUT
  stSettings : ST_NLinkSettings := ( sHost := '127.0.0.1', nPort := 2002 ); (* Link/
connection parameter/settings *)
  nOpt       : DWORD := 0; (* Additional options (reserved) *)
  bExecute   : BOOL; (* Rising edge at this input starts command execution *)
  tTimeout   : TIME := T#45s; (* Maximum time allowed for the execution of the command *)
END_VAR
```

stSettings: [Structure](#) [▶ 21] with configuration settings for the communication connection. The member variables of this structure should contain parameters such as port number, host name, etc.

nOpt: Reserved for future applications.

bExecute: The block is activated by a rising edge at this input.

tTimeout: Specifies the timeout that must not be exceeded on receipt of the command.

VAR_OUTPUT

```
VAR_OUTPUT
  bBusy   : BOOL;
  bError  : BOOL;
  nErrID  : UDINT;
END_VAR
```

bBusy: When the function block is activated this output is set. It remains set until feedback is received. While Busy = TRUE, no new command will be accepted at the inputs. Please note that it is not the command execution but the receipt of the command that is monitored.

bError: In the event of an error during the command transfer, this output is set once the *bBusy* output has been reset.

nErrID: Supplies the [error number](#) [▶ 24] when the *bError* output is set.

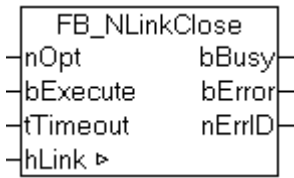
Example:

See [examples](#) [▶ 28].

Requirements

Development environment	Target platform	PLC libraries to be linked
TwinCAT v2.11.0 Build > 1536	PC or CX (x86, ARM)	TcS5S7Comm.Lib

5.2 FB_NLinkClose



This function block closes the network connection to the third-party system.

VAR_IN_OUT

```
VAR_IN_OUT
    hLink : T_HNLINK;
END_VAR
```

hLink : Network connection handle [▶ 22]. This is where the instance of the variable used for establishing the connection by the FB_NLinkOpen [▶ 14] function block is transferred.

VAR_INPUT

```
VAR_INPUT
    nOpt      : DWORD := 0; (* Additional options (reserved) *)
    bExecute  : BOOL;  (* Rising edge at this input starts command execution *)
    tTimeout  : TIME := DEFAULT_ADS_TIMEOUT;
    (* Maximum time allowed for the execution of the command *)
END_VAR
```

nOpt: Reserved for future applications.

bExecute: The block is activated by a rising edge at this input.

tTimeout: Specifies the timeout that must not be exceeded on receipt of the command.

VAR_OUTPUT

```
VAR_OUTPUT
    bBusy     : BOOL;
    bError    : BOOL;
    nErrID    : UDINT;
END_VAR
```

bBusy: When the function block is activated this output is set. It remains set until a feedback is received. While Busy = TRUE, no new command will be accepted at the inputs. Please note that it is not the command execution but the receipt of the command that is monitored.

bError: In the event of an error during the command transfer, this output is set once the *bBusy* output has been reset.

nErrID: Supplies the error number [▶ 24] when the *bError* output is set.

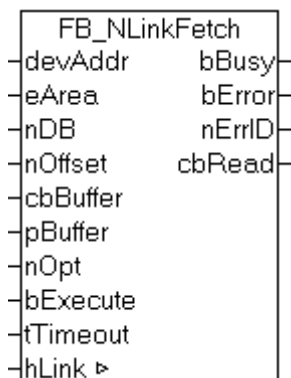
Example:

See examples [▶ 28].

Requirements

Development environment	Target platform	PLC libraries to be linked
TwinCAT v2.11.0 Build > 1536	PC or CX (x86, ARM)	TcS5S7Comm.Lib

5.3 FB_NLinkFetch



This function block reads data from the third-party system and writes them to the local TwinCAT system (FETCH service).

VAR_IN_OUT

```
VAR_IN_OUT
  hLink : T_HNLINK;
END_VAR
```

hLink : [Network connection handle](#) [► 22]. This is where the instance of the variable used for establishing the connection by the [FB_NLinkOpen](#) [► 14] function block is transferred.

VAR_INPUT

```
VAR_INPUT
  devAddr      : BYTE := 2;
  (* Device address, MPI address of the remote component or its PROFIBUS address *)
  eArea        : E_NLinkArea := eNLinkArea_DB; (* PLC data area *)
  nDB          : WORD := 0; (* Data block number from which the data is read (DB, DX only). *)
  nOffset      : WORD := 0; (* Start byte/word address from which the data are taken to *)
  cbBuffer     : UDINT(0..MAX_NLink_DATALENGTH) := 0; (* Contains the max. number of destination by
  tes to be received *)
  pBuffer      : DWORD := 0; (* Contains the address of the destination buffer for the received dat
  a *)
  nOpt         : DWORD := 0; (* Additional options (reserved) *)
  bExecute     : BOOL; (* Rising edge at this input starts command execution *)
  tTimeout     : TIME := DEFAULT_ADS_TIMEOUT;
  (* Maximum time allowed for the execution of the command *)
END_VAR
```

devAddr: Device address, MPI address of the component on the third-party system.

eArea: [PLC data area](#) [► 20] in the third-party system (inputs, outputs, data block etc.) from which data are to be read.

nDB: Data block number in the third-party system. Data are to be read from this data block (only relevant for access to data blocks and extended data blocks).

nOffset: The start byte address or word address from which data are to be read in the third-party system. Whether a byte or word address is to be specified depends on the type of external controller/protocol used and the data area to be accessed. Further information can be found here: [Description of the PLC data area parameter values](#) [► 12].

cbBuffer: Maximum number of data bytes to be read. The byte length is specified even for access to word addresses.

pBuffer: Pointer/address of the target data buffer in the TwinCAT system. The read data are written into this buffer. The address can be determined with the ADR operator. The target data buffer must not be smaller than the specified length of the data to be read.

nOpt: Reserved for future applications.

bExecute: The block is activated by a rising edge at this input.

tTimeout: Specifies the timeout that must not be exceeded on receipt of the command.

VAR_OUTPUT

```
VAR_OUTPUT
  bBusy   : BOOL;
  bError  : BOOL;
  nErrID  : UDINT;
  cbRead  : UDINT := 0; (* Number of recend realy returned data bytes *)
END_VAR
```

bBusy: When the function block is activated this output is set. It remains set until a feedback is received. While Busy = TRUE, no new command will be accepted at the inputs. Please note that it is not the command execution but the receipt of the command that is monitored.

bError: In the event of an error during the command transfer, this output is set once the *bBusy* output has been reset.

nErrID: Supplies the error number [▶ 24] when the *bError* output is set.

cbRead: Number of successfully read data bytes.

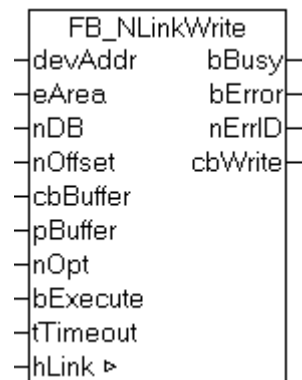
Example:

See examples [▶ 28].

Requirements

Development environment	Target platform	PLC libraries to be linked
TwinCAT v2.11.0 Build > 1536	PC or CX (x86, ARM)	TcS5S7Comm.Lib

5.4 FB_NLinkWrite



This function block sends data from the local TwinCAT system to the third-party system (WRITE service).

VAR_IN_OUT

```
VAR_IN_OUT
  hLink : T_HNLINK;
END_VAR
```

hLink : Network connection handle [▶ 22]. This is where the instance of the variable used for establishing the connection by the FB_NLinkOpen [▶ 14] function block is transferred.

VAR_INPUT

```
VAR_INPUT
  devAddr      : BYTE := 2;
  (* Device address, MPI address of the remote component or its PROFIBUS address *)
  eArea       : E_NLinkArea := eNLinkArea_DB; (* PLC data area *)
  nDB         : WORD := 0; (* Data block number to which the data is written (DB, DX only) *)
  nOffset     : WORD := 0; (* Start byte/word address to which the data is send *)
  cbBuffer    : UDINT(0..MAX_NLink_DATALENGTH) := 0;
```

```
(* Contains the number of source bytes to be send *)
  pBuffer      : DWORD := 0; (* Contains the address of the source buffer containing the data to be
send *)
  nOpt        : DWORD := 0; (* Additional options (reserved) *)
  bExecute    : BOOL; (* Rising edge at this input starts command execution *)
  tTimeout    : TIME := DEFAULT_ADS_TIMEOUT;
(* Maximum time allowed for the execution of the command *)
END_VAR
```

devAddr: Device address, MPI address of the component on the third-party system.

eArea: [PLC data area](#) [► 20] in the third-party system (inputs, outputs, data block, etc.) to which data are to be written.

nDB: Data block number in the third-party system. Block in which data are to be written (only relevant for access to data blocks and extended data blocks).

nOffset: The start byte address or word address from which data are to be written to the third-party system. Whether a byte or word address is to be specified depends on the type of external controller/protocol used and the data area to be accessed. Further information can be found here: [Description of the PLC data area parameter values](#) [► 12].

cbBuffer: Maximum byte length of the data to be transferred from the source data buffer in the local TwinCAT system to the third-party system. The byte length is specified even for access to word addresses.

pBuffer: Pointer/address for the source data buffer. Data from this buffer are transferred to the third-party system. The address can be determined with the ADR operator. The source data buffer must not be smaller than the specified length of the data to be sent.

nOpt: Reserved for future applications.

bExecute: The block is activated by a rising edge at this input.

tTimeout: Specifies the timeout that must not be exceeded on receipt of the command.

VAR_OUTPUT

```
VAR_OUTPUT
  bBusy      : BOOL;
  bError     : BOOL;
  nErrID    : UDINT;
  cbWrite   : UDINT := 0; (* Number of successfull written data bytes *)
END_VAR
```

bBusy: When the function block is activated this output is set. It remains set until a feedback is received. While Busy = TRUE, no new command will be accepted at the inputs. Please note that it is not the command execution but the receipt of the command that is monitored.

bError: In the event of an error during the command transfer, this output is set once the *bBusy* output has been reset.

nErrID: Supplies the [error number](#) [► 24] when the *bError* output is set.

cbWrite: In case of success: Number of successfully written data bytes.

Example:

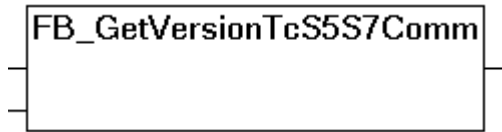
See [examples](#) [► 28].

Requirements

Development environment	Target platform	PLC libraries to be linked
TwinCAT v2.11.0 Build > 1536	PC or CX (x86, ARM)	TcS5S7Comm.Lib

6 Functions

6.1 F_GetVersionTcS5S7Comm



This function can be used to read PLC library version information.

FUNCTION F_GetVersionTcS5S7Comm : UINT

```
VAR_INPUT
    nVersionElement : INT;
END_VAR
```

nVersionElement : Version element to be read. Possible parameters:

- 1 : major number;
- 2 : minor number;
- 3 : revision number;

Requirements

Development environment	Target platform	PLC libraries to be linked
TwinCAT v2.11.0 Build > 1536	PC or CX (x86, ARM)	TcS5S7Comm.Lib

7 Data types

7.1 E_NLinkArea

```

TYPE E_NLinkArea:
(
  eNLinkArea_DB := 16#01, (* Source/dest. data from/to data block in main memory. *)
  eNLinkArea_M := 16#02, (* Source/dest. data from/to flag area. *)
  eNLinkArea_I := 16#03, (* Source/dest. data from/to process image of the inputs. *)
  eNLinkArea_Q := 16#04, (* Source/dest. data from/to process image of the outputs. *)
  eNLinkArea_P := 16#05, (* Source/dest. data from/to in I/O modules (S7 only PI, PQ). *)
  eNLinkArea_C := 16#06, (* Source/dest. data from/to counter cells. *)
  eNLinkArea_T := 16#07, (* Source/dest. data from/to timer cells. *)
  eNLinkArea_DX := 16#0A (* Source/dest. data from/to extended data block. *)
  eNLinkArea_OS := 16#100 (* Read PLC operational status information *)
);
END_TYPE

```

PLC data area (inputs, outputs, flag, data block etc.) to be accessed. The parameter values differ depending on the type of external control used, the adapter or the communication protocol. Some data areas may not be available, or the data have to be read/written word by word instead of byte by byte .

More detailed information on this topic can be found here: [Description of the PLC data area parameter values](#) [► 12].

Requirements

Development environment	Target platform	PLC libraries to be linked
TwinCAT v2.11.0 Build > 1536	PC or CX (x86, ARM)	TcS5S7Comm.Lib

7.2 E_NLinkProtocol

```

TYPE E_NLinkProtocol:
(
  eNLinkProtocol_IBH_SX_PP := 0, (* IBH Link protocol (used by IBH Link S5++ adater) *)
  eNLinkProtocol_MPI (* Hilscher NL-MPI protocol (used only by IBH Link S7++ adapter) *)
);
END_TYPE

```

Communication protocol (depends on the hardware used).

Requirements

Development environment	Target platform	PLC libraries to be linked
TwinCAT v2.11.0 Build > 1536	PC or CX (x86, ARM)	TcS5S7Comm.Lib

7.3 E_NLinkTransport

```

TYPE E_NLinkTransport:
(
  eNLinkTransport_TCPIP := 0
);
END_TYPE

```

Transport protocol (determines how the data are to be transported). Only the stream-oriented TCP/IP protocol is supported at present.

Requirements

Development environment	Target platform	PLC libraries to be linked
TwinCAT v2.11.0 Build > 1536	PC or CX (x86, ARM)	TcS5S7Comm.Lib

7.4 E_NLinkOpState

```

TYPE E_NLinkOpState:
(
  eNLinkOpState_STOP := 0,
  eNLinkOpState_START := 1,
  eNLinkOpState_RUN := 2,
  eNLinkOpState_UNKNOWN := 3
);
END_TYPE
    
```

Operational state (only S7);

Requirements

Development environment	Target platform	PLC libraries to be linked
TwinCAT v2.11.0 Build > 1536	PC or CX (x86, ARM)	TcS5S7Comm.Lib

7.5 ST_NLinkDebug

Via these parameters the debug outputs can be activated/deactivated during troubleshooting (standard setting: inactive)

The debug outputs are shown in the log view of the TwinCAT System Manager.

```

TYPE ST_NLinkDebug:
STRUCT
  eTx      : E_DbgDirection := eDbgDirection_OFF; (* Enable/disable debug output of data-requests/
responses. *)
  eRx      : E_DbgDirection := eDbgDirection_OFF; (* Enable/disable debug output of data-indications/
confirmations. *)
  bState   : BOOL := TRUE; (* TRUE => Enable debug output *)
END_STRUCT
END_TYPE
    
```

eTx: Activates/deactivates the debug output for the send buffer.

eRx: Activates/deactivates the debug output for the receive buffer.

bState: Activates/deactivates status messages when establishing or closing the socket connection.

Requirements

Development environment	Target platform	PLC libraries to be linked
TwinCAT v2.11.0 Build > 1536	PC or CX (x86, ARM)	TcS5S7Comm.Lib

7.6 ST_NLinkSettings

```

TYPE ST_NLinkSettings:
STRUCT
  sSrvNetID      : T_AmsNetID := ''; (* The network address of the TcpIpServer.exe. If empty st
ring=>server runs on local system *)
  eProtocol       : E_NLinkProtocol := eNLinkProtocol_IBH_SX_PP; (* Lan link protocol type *)
  eTransport      : E_NLinkTransport := eNLinkTransport_TCPIP; (* Used transport protocol *)
  sHost          : T_IPv4Addr := '127.0.0.1'; (* Remote (server) address. String containing an
(Ipv4) Internet Protocol dotted address. *)
  nPort          : UDINT := 2002; (* Remote (server) Internet Protocol (IP) port. *)
  local_T_selector : STRING := '00 01'; (* local-transport-selector *)
  remote_T_selector : STRING := '00 01'; (* remote-transport-selector *)
  bServer         : BOOL := FALSE; (* Reserved: TRUE = server connection, FALSE = client connect
ion *)
  bThrottle       : BOOL := FALSE; (* If set reduces the number of polling socket read requests
*)
  debug           : ST_NLinkDebug;
END_STRUCT
END_TYPE
    
```

sSrvNetID: TwinCAT network address of the TwinCAT TCP/IP connection server (default = empty string).

eProtocol: the communication protocol to be used is set via this [parameter \[► 20\]](#).

eTransport: the transport protocol to be used is set via this [parameter](#) [► 20].

sHost: string containing the network address (Ipv4) of the communication partner (default = '127.0.0.1').

nPort: Internet Protocol (IP) port (default = 2002).

local_T_selector: local TSAP address (reserved).

remote_T_selector: remote TSAP address (reserved).

bServer: FALSE = client (active) connection. TRUE = server (passive) connection. Currently only client connections are permitted.

bThrottle: if this parameter is set, then the number of polling socket calls is reduced if no or few data are received, i.e. the poll cycle is slowed down and thus the system load is reduced (default = FALSE).

debug: optional debug output [settings](#) [► 21].

Background information on configuring the eProtocol and eTransport parameters:

The IBHsoftec S7++ adapter can theoretically communicate with the S7 hardware via more than one communication path (protocol).

- Via Hilscher MPI protocol + TCP/IP as transport protocol.
- Via IBHsoftec S5++/S7++ protocol + RFC1006 as transport protocol.

The TwinCAT S5/S7 communication library currently supports only the first communication path, because currently only the TCP/IP is supported as transport protocol.

I.e. for the communication with S7 devices please set the following setting:

```
eProtocol := eNLinkProtocol_MPI;
eTransport := eNLinkTransport_TCPIP;
```

For the communication with the S5 devices please set the following setting:

```
eProtocol := eNLinkProtocol_IBH_SX_PP;
eTransport := eNLinkTransport_TCPIP;
```

Requirements

Development environment	Target platform	PLC libraries to include
TwinCAT v2.11.0 Build > 1536	PC or CX (x86, ARM)	TcS5S7Comm.Lib

7.7 T_HNLINK

A variable of this type represents a network connection handle. The structural elements should not be written to or modified directly. Variables of this type are used for internal data exchange.

One instance is required for each network connection (TwinCAT system<->Ethernet LAN adapter<->external controller). The same instance is then transferred to other communication blocks as VAR_IN_OUT parameter.

```
TYPE T_HNLINK :
STRUCT
END_STRUCT
END_TYPE
```

Requirements

Development environment	Target platform	PLC libraries to be linked
TwinCAT v2.11.0 Build > 1536	PC or CX (x86, ARM)	TcS5S7Comm.Lib

8 Constants

Name	Value	Description
MAX_NLINK_BUFFER_SIZE	16#FFFF	Max. internal buffer size for RX/TX (64k)
MAX_NLINK_DATALEN	2048	Max. data length supported by the PLC blocks for transfers. The actual max. length may be smaller due to specific device characteristics.
MAX_NLINK_HEADERLEN	16	Max. length of the frame header for the NL-MPI and IBHNet protocols.
MAX_NLINK_MPI_CONNECTIONS	16	Max. number of open MPI connections. If this number is exceeded, the oldest connections are closed automatically.

Requirements

Development environment	Target platform	PLC libraries to be linked
TwinCAT v2.11.0 Build > 1536	PC or CX (x86, ARM)	TcS5S7Comm.Lib

9 Error codes

Requirements

Codes (hex)	Codes (dec)	Error source	Description
0x00000000-0x00007800	0-30720	TwinCAT system error codes	TwinCAT system error (including ADS error codes)
0x00008000-0x000080FF	32768-33023	Internal TwinCAT TCP/IP Connection Server error codes	Internal error of the TwinCAT TCP/IP Connection Server
0x00008300-0x000085FF	33536-34303	<u>Internal TwinCAT S5/S7 communication error codes</u> [▶ 24]	Internal error in the S5/S7 Ethernet/LAN module or the TwinCAT PLC library
0x80070000-0x8007FFFF	2147942400-2148007935	Error source = Code - 0x80070000 = Win32 system error codes	Win32 system error (including Windows sockets error codes)

9.1 Internal S5/S7 communication error codes

Code (hex)	Code (dec)	Symbolic constant	Description
		General error codes	Offset 0x8300 + error code
0x00000000	0	NLINK_COMMERR_NOERROR	No error.
0x00008300	33536	NLINK_COMMERR_TIMEOUT	Timeout during communication (transaction)
0x00008301	33537	NLINK_COMMERR_TXOVERFLOW	Send buffer overflow
0x00008302	33538	NLINK_COMMERR_RXOVERFLOW	Receive buffer overflow
0x00008303	33539	NLINK_COMMERR_FATAL	Fatal internal error
0x00008304	33540	NLINK_COMMERR_SYSTEMID	Invalid system ID
0x00008305	33541	NLINK_COMMERR_HEADERLEN	Invalid header length parameter
0x00008306	33542	NLINK_COMMERR_OPCODE	Invalid OP code, OP code length or OP code word
0x00008307	33543	NLINK_COMMERR_INVALIDSIZE	Invalid data length
0x00008308	33544	NLINK_COMMERR_INVALIDSTATE	Device or connection is in a faulty/invalid state
0x00008309	33545	NLINK_COMMERR_INVALIDVALUE	An invalid parameter value was specified
0x0000830A	33546	NLINK_COMMERR_INVALIDAREA	Invalid/unsupported data area
0x0000830B	33547	NLINK_COMMERR_UNEXPECTED	Unexpected error
		IBH Link S5++ specific error codes	Offset 0x8400 + error code
0x00008402	33794	NLINK_COMMERR_IBH_AREA_NOT_AVAILABLE	Data area not available in the controller, data block not available
0x00008403	33795	NLINK_COMMERR_IBH_AREA_TOO_SMALL	Requested area is too small
0x00008406	33798	NLINK_COMMERR_IBH_INVALID_HEADER	Invalid frame/header format
0x00008407	33799	NLINK_COMMERR_IBH_DATA_OVERFLOW	Permitted data size exceeded, data buffer too small
0x00008409	33801	NLINK_COMMERR_IBH_TIMEOUT	Timeout during communication with the controller
0x0000840A	33802	NLINK_COMMERR_IBH_DATA_UNDERFLOW	Too few data were received

Code (hex)	Code (dec)	Symbolic constant	Description
		Hilscher NL-MPI interface error codes	Offset 0x8500 + MPI response message error code
0x00008501	34049	NLINK_COMMERR_MPI_UE	Timeout from remote station remote station remote station has not responded within 1 sec.timeout
0x00008502	34050	NLINK_COMMERR_MPI_RR	Resource unavailable remote station remote station has no left buffer space for the requested service
0x00008503	34051	NLINK_COMMERR_MPI_RS	Requested function of master is not activated within the remote station. The connection seems to be closed in the remote station. Try to send command again
0x00008511	34065	NLINK_COMMERR_MPI_NA	No response of the remote station remote station check network wiring, check remote address, check baud rate
0x00008512	34066	NLINK_COMMERR_MPI_DS	Master not into the logical token ring network in general check master DP-Address or highest-station-Addresses of other masters. Examine bus wiring to bus short circuits
0x00008514	34068	NLINK_COMMERR_MPI_LR	Resource of the local FDL controller not available or not sufficient. HOST too many messages. no more segments in DEVICE free
0x00008515	34069	NLINK_COMMERR_MPI_IV	The specified msg.data_cnt parameter invalid HOST check the limit of 222 bytes (read) respectively 216 bytes (write) in msg.data_cnt
0x00008530	34096	NLINK_COMMERR_MPI_TO	Timeout, the request message was accepted but no indication is sent back by the remote station remote station MPI protocol error, or station not presenter
0x00008539	34105	NLINK_COMMERR_MPI_SE	Sequence fault, internal state machine error. Remote station does not react like awaited or a reconnection was retried while connection is already open or device has no SAPs left to open connection channel
0x00008585	34181	NLINK_COMMERR_MPI_REJ_IV	Specified offset address out of limits or not known in the remote station HOST please check msg.data_adr if present or offset parameter in request message

Code (hex)	Code (dec)	Symbolic constant	Description
0x00008586	34182	NLINK_COMMERR_MPI_REJ_PDU	Wrong PDU coding in the MPI response of the remote station DEVICE contact hotline
0x00008587	34183	NLINK_COMMERR_MPI_REJ_OP	Specified length to write or to read results in an access outside the limits HOST please check msg.data_cnt length in request message
0x00008588	34184	NLINK_COMMERR_MPI_REJ_OP	Specified address not defined in the remote station, please check msg.data_adr in the request message
0x00008589	34185	NLINK_COMMERR_MPI_REJ_OP	MPI remote station not in the right operational mode, bring S7 into RUN-P Mode

Requirements

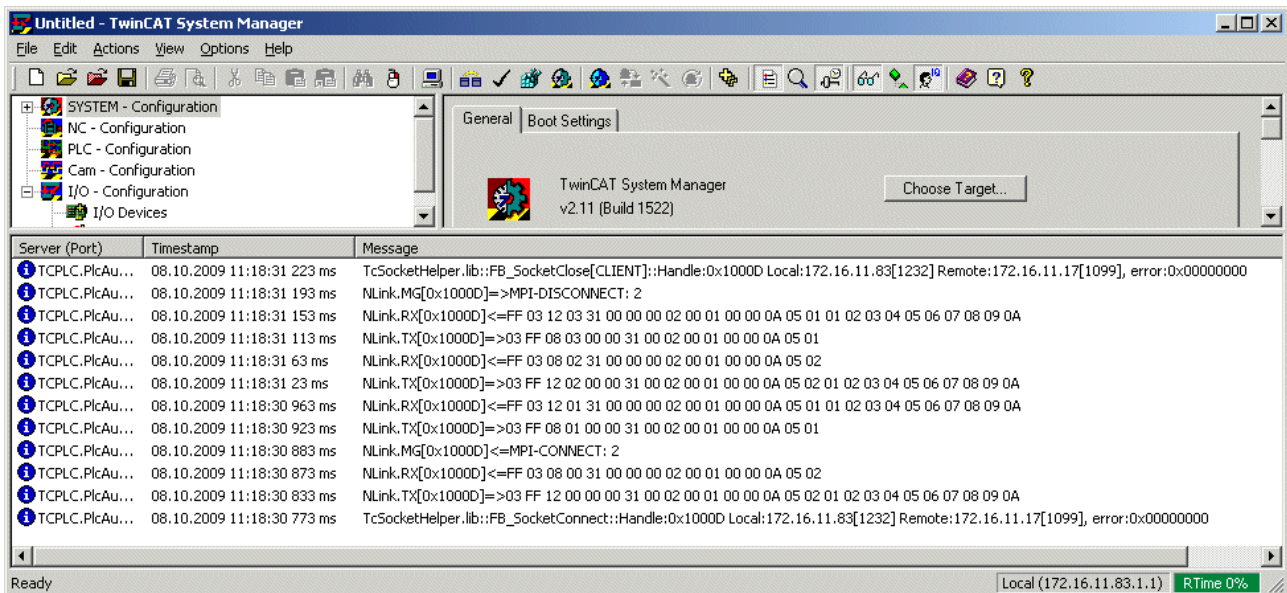
Development environment	Target platform	PLC libraries to be linked
TwinCAT v2.11.0 Build > 1536	PC or CX (x86, ARM)	TcS5S7Comm.Lib

9.2 Troubleshooting/diagnostics

1. In the event of connection problems the PING command can be used to ascertain whether the external controller can be reached via the network connection. If this is not the case, check the configuration/power supply and the network connection of the adapter module. Instructions for configuring the adapter module can be found in the manufacturer documentation for the adapter.
2. Sniffer tools such as Wireshark enable logging of the entire network communication. The log can then be analysed by Beckhoff support staff.
3. Check the hardware and software requirements described in this documentation (adapter manufacturer, model, firmware version).
4. If you access the external controller via the IBH Link S5++ or IBH Link S7++ adapter in conjunction with TCP/IP as transport protocol, no connections have to be configured/parameterised in the external controller. In the event of data access problems, check whether the corresponding data areas (flags, data blocks etc.) in the external controller are configured correctly and available.
5. Check the input parameters that are transferred to the function blocks (network address, MPI address, data block number etc.) for correctness. Check whether the function block issues an error code. The documentation for the error codes can be found here: [Overview of error codes](#) [▶ 24].
6. A simple debug output is integrated in the TwinCAT PLC library. This must be activated before establishing the connection at the [FB NlinkOpen](#) [▶ 14] function block. Open the TwinCAT System Manager and activate the LogView window. The following image shows some debug output strings.

Example:

Communication via the IBH Link S7++ (NL-MPI protocol via TCP/IP). Only complete frames are logged (16 byte header + PLC data from the corresponding data area), i.e. byte 15 is followed by the read and written PLC data.



Some explanations regarding the messages:

No	Message	Meaning
12	TcSocketHelper.lib::FB_SocketClose[CLIENT]::Handle: 0x1000D Local:172.16.11.83[1232] Remote:172.16.11.17[1099], error:0x00000000	TCP/IP connection (internal handle: 0x100D) is closed error-free.
11	NLink.MG[0x1000D]>=>MPI-DISCONNECT: 2	The internal MPI connection to the device with MPI address 2 is closed.
10	NLink.RX[0x1000D]<=FF 03 12 03 31 00 00 00 02 00 01 00 00 0A 05 01 01 02 03 04 05 06 07 08 09 0A	READ response was received.
9	NLink.TX[0x1000D]>=>03 FF 08 03 00 00 31 00 02 00 01 00 00 0A 05 01	READ request was sent...
8	NLink.RX[0x1000D]<=FF 03 08 02 31 00 00 00 02 00 01 00 00 0A 05 02	WRITE response was received...
7	NLink.TX[0x1000D]>=>03 FF 12 02 00 00 31 00 02 00 01 00 00 0A 05 02 01 02 03 04 05 06 07 08 09 0A	WRITE request was sent...
6	NLink.RX[0x1000D]<=FF 03 12 01 31 00 00 00 02 00 01 00 00 0A 05 01 01 02 03 04 05 06 07 08 09 0A	READ response was received... (10 bytes: 01 02 03 04 05 06 07 08 09 0A received successfully).
5	NLink.TX[0x1000D]>=>03 FF 08 01 00 00 31 00 02 00 01 00 00 0A 05 01	READ request was sent.
4	NLink.MG[0x1000D]<=MPI-CONNECT: 2	The internal MPI connection to the device with MPI address 2 is opened.
3	NLink.RX[0x1000D]<=FF 03 08 00 31 00 00 00 02 00 01 00 00 0A 05 02	WRITE response was received... (only acknowledgement).
2	NLink.TX[0x1000D]>=>03 FF 12 00 00 00 31 00 02 00 01 00 00 0A 05 02 01 02 03 04 05 06 07 08 09 0A	WRITE request was sent... (10 bytes: 01 02 03 04 05 06 07 08 09 0A are written to the external controller).
1	TcSocketHelper.lib::FB_SocketConnect::Handle:0x1000D Local:172.16.11.83[1232] Remote:172.16.11.17[1099], error:0x00000000	TCP/IP connection (internal handle: 0x100D) is established error-free.

10 Samples

The communication parameters in the project examples must be set to match your configuration.

PLC project	Third-party system	Adapter	Communication parameters	Data area parameter values	Description
https://infosys.beckhoff.com/content/1033/tcplclibs5s7com/Resourcen/11426458507/.zip	S5	Ethernet adapter: IBH Link S5++	<ul style="list-style-type: none"> • Protocol: IBH S5 LAN Link protocol • Transport: TCP/IP • IP address: '172.16.11.16' • Port number: 2002 	<ul style="list-style-type: none"> • Data area: DB • Data block number: 16 • Write 1 word (2 bytes) from the start address (word offset): 9 • Read 10 words (20 bytes) from the start address (word offset): 0 	The data in the data block are accessed word by word.
https://infosys.beckhoff.com/content/1033/tcplclibs5s7com/Resourcen/11426459915/.zip	S7	Ethernet adapter: IBH Link S7++	<ul style="list-style-type: none"> • Protocol: NetLink MPI protocol (NL-MPI) • Transport: TCP/IP • IP address: '172.16.11.17' • Port number: 1099 • MPI address: 2 	<ul style="list-style-type: none"> • Data area: DB • Data block number: 1 • Write 10 bytes from the start address (byte offset): 0 • Read 10 bytes from the start address (byte offset): 0 	<p>The data in the data block are accessed byte by byte.</p> <p>In addition to the selected data area the operating status of the controller is read (STOP, RUN, START...).</p>

More Information:

www.beckhoff.com/de-de/produkte/automation/twincat/tsxxx-twincat-2-supplements/

Beckhoff Automation GmbH & Co. KG
Hülshorstweg 20
33415 Verl
Germany
Phone: +49 5246 9630
info@beckhoff.com
www.beckhoff.com

