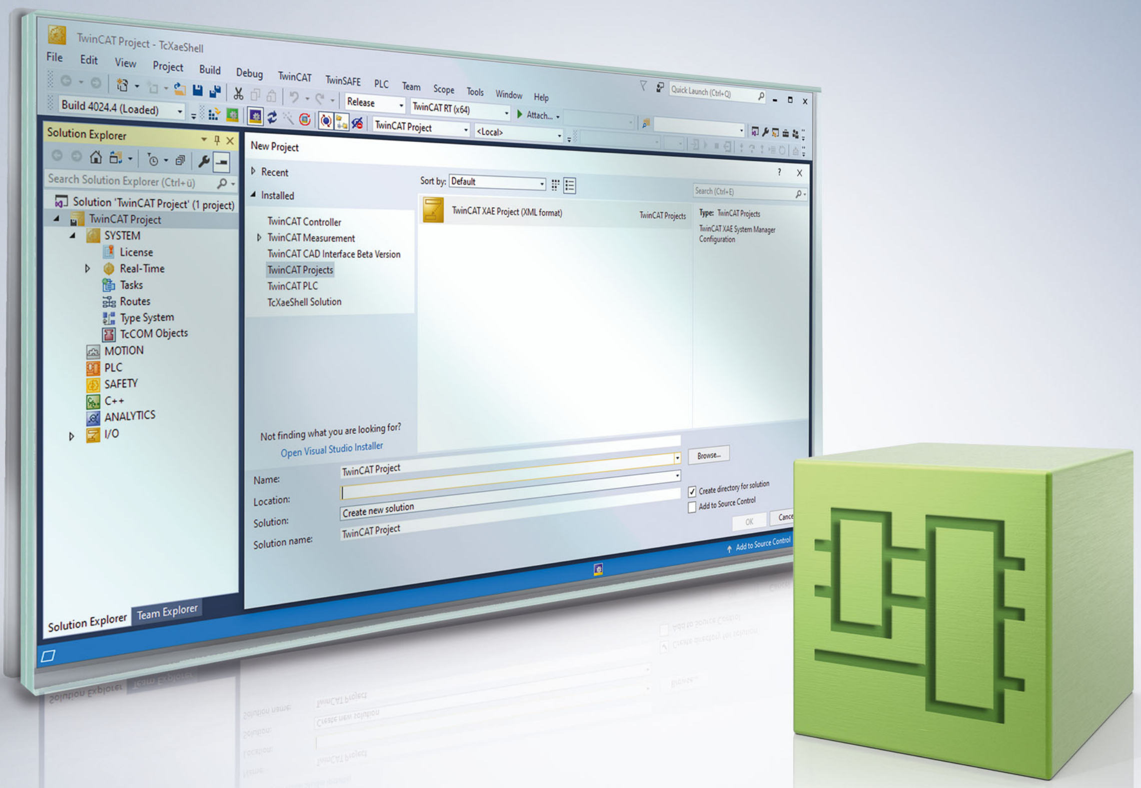
BECKHOFF New Automation Technology

Manual | EN

TE1000

TwinCAT 3 | PLC Library: Tc2_EIB



2024-04-23 | Version: 1.15.1

Table of contents

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

Copyright

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

ℹ 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 Introduction

The Tc2_EIB library is a TwinCAT PLC library for data exchange with EIB devices.

This library should only be used in conjunction with a KL6301 (EIB master terminal).

The user of this library requires basic knowledge of the following:

- TwinCAT XAE
- PC and network knowledge
- Structure and properties of the Beckhoff Embedded PC and its Bus Terminal system
- Technology of EIB devices
- Relevant safety regulations for building technical equipment

This software library is intended for building automation system partners of Beckhoff Automation GmbH & Co. KG. The system partners operate in the field of building automation and are concerned with the installation, commissioning, expansion, maintenance and service of measurement, control and regulating systems for the technical equipment of buildings.

The Tc2_EIB library is usable on all hardware platforms that support TwinCAT 3.1 or higher.

Hardware documentation in the Beckhoff information system:

https://infosys.beckhoff.com/content/1033/kl6301/index.html

3 EIB

Working with the EIB bus terminal requires function blocks, which are described in this documentation.

Three different modes are available from firmware version B1 and library V3, which can be enabled in the KL6301 function block.

Mode 0: 4 filters with 64 group addresses each (compatible with firmware B0). Acknowledgement with ACK to group addresses within the filters.

Mode 1: 8 filters with 32 group addresses each. Acknowledgement with ACK to group addresses within the filters.

Mode 2: 8 filters with 32 group addresses each, but inverted. Data reception only from group addresses outside the filters. Acknowledgement with ACK to group addresses outside the filters.

Mode 100: Reception of all telegrams of each group address (monitor function). No data transmission possible. No acknowledgement with ACK.

Sending

The KL6301 sends data individually. This means that a Data variable sent to the KL6301 is sent to the EIB network individually. Subsequent EIB data can only be transferred to the KL6301 after a successful transfer. Two types of EIB telegrams can be sent:

- WRITE_GROUP for writing data to other EIB devices
- READ_GROUP_REQ for requesting data from other EIB devices

Receiving

The KL6301 has a maximum of 8 filter addresses. These filters filter the EIB group addresses. Only EIB telegrams entered in the filter are visible in the process image and are acknowledged.

A filter may contain up to 64 group addresses. With 4 filters multiplied with 64 entries a total of 256 group addresses are available. With 8 filters multiplied with 32 entries a total of 256 group addresses are available. The system is configured via a function block. The group addresses are loaded and are immediately active when the bus terminal is initialized.

At least one filter has to be parameterized. The data type is not significant for the filter setting.

Monitor function

No filters must be set if mode 100 is enabled. The filters EIB_GROUP_FILTER are simply left empty and not written.

4 Programming

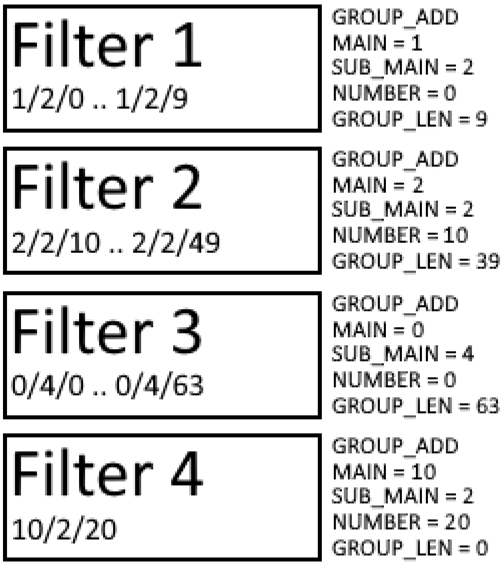
4.1 EIB group filter

The EIB group filters have to be parameterized before the KL6301 can enter data exchange mode. The filters are required for all data with a group address sent to the KL6301. Each group telegram that is also included in the filters is acknowledged and entered in the process data, i.e. it is visible in the function blocks. The KL6301 discards EIB telegrams with group addresses that are not included in the filter.

Example

Filter 1 Group address 1/2/0 Length: 20
All EIB telegrams with group address 1/2/0 - 1/2/19 pass the filter

At least one filter must always be activated.
The selected mode determines the number and length of the group filters. The length specification starts at 0, which corresponds to exactly one entry.


Filter 1
1/2/0 .. 1/2/9
GROUP_ADD
MAIN = 1
SUB_MAIN = 2
NUMBER = 0
GROUP_LEN = 9

Filter 2
2/2/10 .. 2/2/49
GROUP_ADD
MAIN = 2
SUB_MAIN = 2
NUMBER = 10
GROUP_LEN = 39

Filter 3
0/4/0 .. 0/4/63
GROUP_ADD
MAIN = 0
SUB_MAIN = 4
NUMBER = 0
GROUP_LEN = 63

Filter 4
10/2/20
GROUP_ADD
MAIN = 10
SUB_MAIN = 2
NUMBER = 20
GROUP_LEN = 0

Changes for Firmware B1 and library version V3

With firmware version B1 and TwinCAT library Tc2_EIB (V3) instead of **4 filters also 8 filters** can be parameterized. But the maximum length of each filter is reduced from 64 to 32 entries per filter group. So the maximum of all group filters to be received is constant with 256.

4.2 POUs

General

Contents	Description
KL6301 [▸ 14]	Communication with a KL6301
KL6301_EX [▸ 16]	Communication with a KL6301

Read

Function blocks	Description
EIB_2OCTET_FLOAT_REC [▸ 18]	Receiving of 2-byte Float EIB data and conversion to REAL
EIB_2OCTET_SIGN_REC [▸ 19]	Receiving of 2-byte Sign EIB data and conversion to INT
EIB_2OCTET_UNSIGN_REC [▸ 20]	Receiving of 2-byte Unsign EIB data and conversion to UINT
EIB_3BIT_CONTROL_REC [▸ 20]	Receiving of a "3 Bit Controlled" data type
EIB_4OCTET_FLOAT_REC [▸ 21]	Receiving of 4-byte Float EIB data and conversion to REAL
EIB_4OCTET_SIGN_REC [▸ 22]	Receiving of 4-byte Sign EIB data and conversion to DINT
EIB_4OCTET_UNSIGN_REC [▸ 22]	Receiving of 4-byte Unsign EIB data and conversion to UDINT
EIB_8BIT_SIGN_REC [▸ 23]	Receiving of 8 BIT EIB data and conversion to INT
EIB_8BIT_UNSIGN_REC [▸ 24]	Receiving of 8 BIT EIB data and conversion to BYTE
EIB_ALL_DATA_TYPES_REC [▸ 25]	Receives any EIB data
EIB_ALL_DATA_TYPES_REC_EX [▸ 25]	Receives any EIB data
EIB_BIT_CONTROL_REC [▸ 26]	Receiving of a "1 Bit Controlled" data type
EIB_BIT_REC [▸ 27]	Receiving of 1 BIT EIB data and conversion to BOOL
EIB_DATE_REC [▸ 28]	Receiving a date
EIB_TIME_REC [▸ 28]	Receiving a time

Send

Function blocks	Description
EIB_2OCTET_FLOAT_SEND [▶ 30]	Sending a REAL value (conversion to 2 byte Float EIB)
EIB_2OCTET_FLOAT_SEND_EX [▶ 32]	Sending a REAL value (conversion to 2 byte Float EIB)
EIB_2OCTET_SIGN_SEND [▶ 33]	Sending an INT value (conversion to 2 byte Sign EIB)
EIB_2OCTET_SIGN_SEND_EX [▶ 34]	Sending an INT value (conversion to 2 byte Sign EIB)
EIB_2OCTET_UNSIGN_SEND [▶ 35]	Sending an UINT value (conversion to 2 byte Unsign EIB)
EIB_2OCTET_UNSIGN_SEND_EX [▶ 37]	Sending an UINT value (conversion to 2 byte Unsign EIB)
EIB_3BIT_CONTROL_SEND [▶ 38]	Sending a "3 bit Controlled" data type
EIB_3BIT_CONTROL_SEND_EX [▶ 39]	Sending a "3 bit Controlled" data type
EIB_4OCTET_FLOAT_SEND [▶ 40]	Sending a REAL value (conversion to 4 byte Float EIB)
EIB_4OCTET_FLOAT_SEND_EX [▶ 42]	Sending a REAL value (conversion to 4 byte Float EIB)
EIB_4OCTET_SIGN_SEND [▶ 43]	Sending a DINT value (conversion to 4 byte Sign EIB)
EIB_4OCTET_SIGN_SEND_EX [▶ 44]	Sending a DINT value (conversion to 4 byte Sign EIB)
EIB_4OCTET_UNSIGN_SEND [▶ 45]	Sending a UDINT value (conversion to 4 byte Unsign EIB)
EIB_4OCTET_UNSIGN_SEND_EX [▶ 47]	Sending a UDINT value (conversion to 4 byte Unsign EIB)
EIB_8BIT_SIGN_SEND [▶ 48]	Sending an INT value (conversion to 1 byte Sign EIB)
EIB_8BIT_SIGN_SEND_EX [▶ 49]	Sending an INT value (conversion to 1 byte Sign EIB)
EIB_8BIT_UNSIGN_SEND [▶ 51]	Sending a BYTE value (conversion to 1 byte Unsign EIB)
EIB_8BIT_UNSIGN_SEND_EX [▶ 52]	Sending a BYTE value (conversion to 1 byte Unsign EIB)
EIB_ALL_DATA_TYPES_SEND [▶ 54]	Sending any EIB data
EIB_BIT_CONTROL_SEND [▶ 56]	Sending a "1 bit Controlled" data type
EIB_BIT_CONTROL_SEND_EX [▶ 58]	Sending a "1 bit Controlled" data type
EIB_BIT_SEND [▶ 59]	Sending a BOOL value
EIB_BIT_SEND_EX [▶ 60]	Sending a BOOL value
EIB_BIT_SEND_MANUAL [▶ 61]	Sending a BOOL value
EIB_DATE_SEND [▶ 62]	Sending a date
EIB_DATE_SEND_EX [▶ 63]	Sending a date
EIB_READ_SEND [▶ 65]	Sending a *Read_Group_Req*
EIB_TIME_SEND [▶ 66]	Sending a time
EIB_TIME_SEND_EX [▶ 67]	Sending a time

Functions

Function blocks	Description
F_CONV_2GROUP_TO_3GROUP [▶ 69]	Conversion of a 2-stage group address to a 3-stage group address
F_CONV_3GROUP_TO_2GROUP [▶ 69]	Conversion of a 3-stage group address to a 2-stage group address

4.2.1 Function blocks details

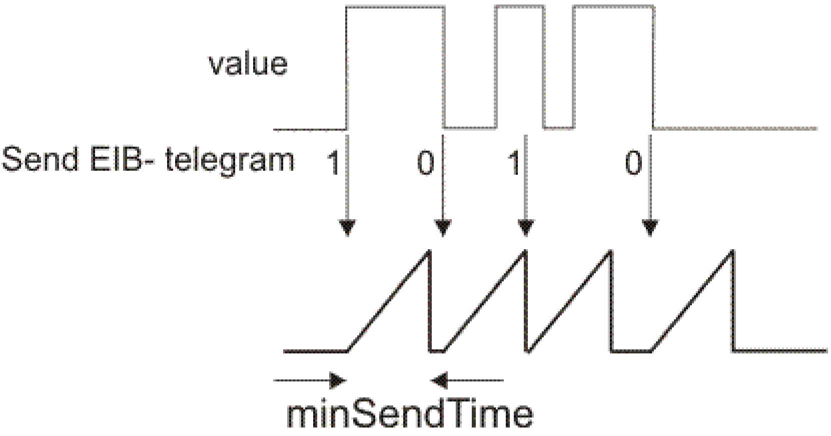
Description	_Rec	_Send				
		_Send	First Cycle	Delta, min. Send Time	Polling	Auto/manual
EIB_BIT	yes [▸ 27]	yes [▸ 59]	no	200 msec	no	Auto
EIB_BIT_EX	no	yes [▸ 60]	yes	1 sec, variable	10 sec, variable	Auto/manual
EIB_BIT_MANUAL	no	yes [▸ 61]	no	no	no	Manual
EIB_BIT_CONTROL	yes [▸ 26]	yes [▸ 56]	no	200 msec	no	Auto
EIB_3BIT_CONTROL	yes [▸ 20]	yes [▸ 38]	no	200 msec	no	Auto
EIB_8BIT_SIGN	yes [▸ 23]	yes [▸ 48]	no	1 sec	no	Auto
EIB_8BIT_SIGN_EX	no	yes [▸ 49]	yes	1 sec, variable	500 msec, variable	Auto/manual
EIB_8BIT_UNSIGN	yes [▸ 24]	yes [▸ 51]	no	1 sec	no	Auto
EIB_8BIT_UNSIGN_EX	no	yes [▸ 52]	yes	1 sec, variable	500 msec, variable	Auto/manual
EIB_2OCTET_SIGN	yes [▸ 19]	yes [▸ 33]	no	1 sec	no	Auto
EIB_2OCTET_SIGN_EX	no	yes [▸ 34]	yes	1 sec, variable	500 msec, variable	Auto/manual
EIB_2OCTET_UNSIGN	yes [▸ 20]	yes [▸ 35]	no	1 sec	no	Auto
EIB_2OCTET_UNSIGN_EX	no	yes [▸ 37]	yes	1 sec, variable	500 msec, variable	Auto/manual
EIB_2OCTET_FLOAT	yes [▸ 18]	yes [▸ 30]	no	1 sec	no	Auto
EIB_2OCTET_FLOAT_EX	no	yes [▸ 32]	yes	1 sec, variable	500 msec, variable	Auto/manual
EIB_TIME	yes [▸ 28]	yes [▸ 66]	yes	no	5 min	Auto
EIB_DATE	yes [▸ 28]	yes [▸ 62]	yes	no	5 min	Auto
EIB_4OCTET_SIGN	yes [▸ 22]	yes [▸ 43]	no	1 sec	no	Auto
EIB_4OCTET_SIGN_EX	no	yes [▸ 44]	yes	1 sec, variable	500 msec, variable	Auto/manual
EIB_4OCTET_UNSIGN	yes [▸ 22]	yes [▸ 45]	no	1 sec	no	Auto
EIB_4OCTET_FLOAT	yes [▸ 21]	yes [▸ 40]	no	1 sec	no	Auto
EIB_4OCTET_FLOAT_EX	no	yes [▸ 42]	yes	1 sec, variable	10 min, variable	Auto/manual
EIB_READ	no	yes [▸ 65]	no	no	no	Manual
EIB_ALL_DATA_TYPES	yes [▸ 25]	yes [▸ 54]	no	1 sec, variable	100 msec, variable	Auto/manual
EIB_ALL_DATA_TYPES_EX	yes [▸ 25]	no	no	no	no	no

_Rec: yes - receiving is supported, no - receiving is not supported

_Send: yes - sending is supported, no - sending is not supported

First Cycle: An EIB telegram is sent when the function block is called for the first time.

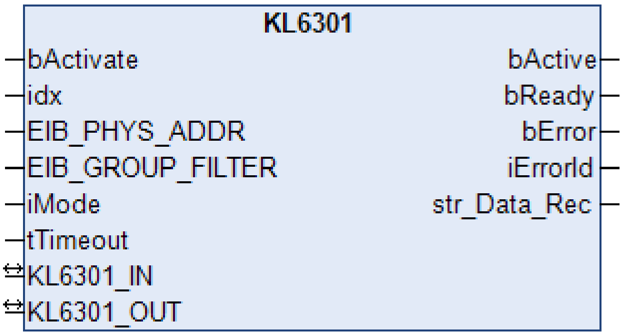
Delta, min. Send Time: An EIB telegram is only sent when the data is changed. The parameter "min. Send Time" is used to activate a "send filter". It does not extend the response time for the first relative change in input, but is active for subsequent changes. The "min. Send Time" describes the minimum time that must be waited between the sending of two telegrams. The "min. Send Time" reduces bus communication and ensures that send instructions from other function blocks can also transmit their EIB telegrams.



Polling: The data are automatically sent at the specified interval, even if the data did not change

Auto/Manual: Auto - Data is sent automatically when the function block is called, Manual - Data is only sent when requested by the function block.

4.2.2 KL6301



This function block deals with the KL6301 EIB Bus Terminal communication. It is used for configuring the KL6301 and for starting the data exchange with the EIB network.

Restrictions

- Only one call per instance
- Call must be made once per PLC cycle
- Instance must be called in the same PLC task as the send and receive blocks assigned to it
- Maximum 64 instances per PLC project allowed

Inputs

```
VAR_INPUT
  bActivate        : BOOL;
  idx              : INT := 1;
  EIB_PHYS_ADDR    : EIB_PHYS_ADDR;
  EIB_GROUP_FILTER : ARRAY [1..8] OF EIB_GROUP_FILTER;
  iMode            : INT;
  tTimeout         : TIME := T#5s;
END_VAR
```

Name	Type	Description
bActivate	BOOL	Activates the function block that configures the KL6301 and then activates the data exchange. If FALSE, any tasks that are still running will be completed and the data exchange will be terminated. If the outputs *bActive* and *bReady* are then FALSE, the function block can be reactivated.
idx	INT	If more than one bus terminal per PLC program is used, each KL6301 must have a unique idx number. Valid values from 1...64.
EIB_PHYS_ADDR	EIB_PHYS_ADDR [▶ 75]	Physical EIB address. The default address is 1.2.3. This address must be unique in the EIB network!
EIB_GROUP_FILTER	ARRAY OF EIB_GROUP_FILTER [▶ 75]	Group address filter. A maximum of 8 filters are possible.
iMode	INT	0 - For firmware B0 and higher - 4 filters with 64 entries each 1 - For firmware B1 and higher - 8 filters with 32 entries each 2 - For firmware B3 and higher - 8 filters, each with 32 inverted entries. In the case of cross communication within the KNX/EIB network of telegrams that are not connected with the KL6301, care must be taken to ensure that these group addresses are contained in the filter so that the terminal does not send an ACK. 100 - For Firmware B1 and higher - Monitor function, all group address telegrams are received. The telegrams are not acknowledged (no ACK is sent). Sending is disabled in this mode.
tTimeout	TIME	Time allowed for a transmit function block to transmit an EIB telegram before a timeout is signaled.

Inputs/outputs

```
VAR_IN_OUT
  KL6301_IN  : ARRAY [1..24] OF BYTE;
  KL6301_OUT : ARRAY [1..24] OF BYTE;
END_VAR
```

Name	Type	Description
KL6301_IN	ARRAY OF BYTE	Is linked with the KL6301 input addresses.
KL6301_OUT	ARRAY OF BYTE	Is linked with the KL6301 output addresses.

Outputs

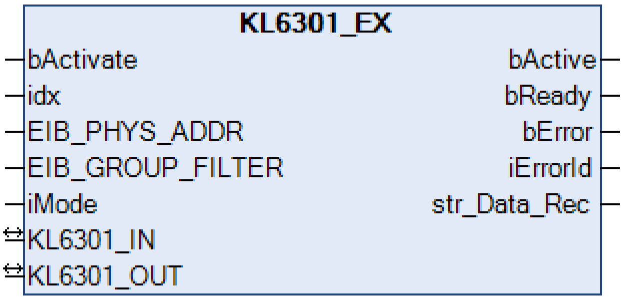
```
VAR_OUTPUT
  bActive      : BOOL;
  bReady       : BOOL;
  bError       : BOOL;
  iErrorId     : EIB_Error_Code;
  str_Data_Rec : EIB_REC;
END_VAR
```

Name	Type	Description
bActive	BOOL	The function block was activated.
bReady	BOOL	The function block is ready for sending and receiving data.
bError	BOOL	This output goes TRUE as soon as an error occurs. This error is described via the *iErrorId* variable.
iErrorId	EIB_ERROR_CODE [▶ 72]	**iErrorId:** The output issues an error code in the event of an error. *bError* goes TRUE at the same time.
str_Data_Rec	EIB_REC [▶ 76]	Connected with the send and receive function blocks.

Requirements

Development environment	required TC3 PLC library
TwinCAT from v3.1.4020.14	Tc2_EIB from 3.3.4.0

4.2.3 KL6301_EX



This function block deals with the KL6301 EIB Bus Terminal communication. It is used for configuring the KL6301 and for starting the data exchange with the EIB network.

BETA: ETS support for search and LED flashing.

Restrictions

- Only one call per instance
- Call must be made once per PLC cycle
- Instance must be called in the same PLC task as the send and receive blocks assigned to it
- Maximum 64 instances per PLC project allowed

Inputs

```
VAR_INPUT
  bActivate        : BOOL;
  idx              : INT := 1;
  EIB_PHYS_ADDR    : EIB_PHYS_ADDR;
  EIB_GROUP_FILTER : ARRAY [1..8] OF EIB_GROUP_FILTER;
  iMode            : INT;
END_VAR
```

Name	Type	Description
bActivate	BOOL	Activates the function block that configures the KL6301 and then activates the data exchange. If FALSE, any tasks that are still running will be completed and the data exchange will be terminated. If the outputs *bActive* and *bReady* are then FALSE, the function block can be reactivated.
idx	INT	If more than one bus terminal per PLC program is used, each KL6301 must have a unique idx number. Valid values from 1...64.
EIB_PHYS_ADDR	EIB_PHYS_ADDR [▶ 75]	Physical EIB address. The default address is 1.2.3. This address must be unique in the EIB network!
EIB_GROUP_FILTER	ARRAY OF EIB_GROUP_FILTER [▶ 75]	Group address filter. A maximum of 8 filters are possible.
iMode	INT	0 - For firmware B0 and higher - 4 filters with 64 entries each 1 - For firmware B1 and higher - 8 filters with 32 entries each 2 - For firmware B3 and higher - 8 filters, each with 32 inverted entries. In the case of cross communication within the KNX/EIB network of telegrams that are not connected with the KL6301, care must be taken to ensure that these group addresses are contained in the filter so that the terminal does not send an ACK. 100 - For Firmware B1 and higher - Monitor function, all group address telegrams are received. The telegrams are not acknowledged (no ACK is sent). Sending is disabled in this mode.

Inputs/outputs

```
VAR_IN_OUT
  KL6301_IN   : ARRAY [1..24] OF BYTE;
  KL6301_OUT  : ARRAY [1..24] OF BYTE;
END_VAR
```

Name	Type	Description
KL6301_IN	ARRAY OF BYTE	Is linked with the KL6301 input addresses.
KL6301_OUT	ARRAY OF BYTE	Is linked with the KL6301 output addresses.

Outputs

```
VAR_OUTPUT
  bActive      : BOOL;
  bReady       : BOOL;
  bError       : BOOL;
  iErrorId     : EIB_Error_Code;
  str_Data_Rec : EIB_REC;
END_VAR
```

Name	Type	Description
bActive	BOOL	The function block was activated.
bReady	BOOL	The function block is ready for sending and receiving data.
bError	BOOL	This output goes TRUE as soon as an error occurs. This error is described via the *iErrorId* variable.
iErrorId	EIB_ERROR_CODE [▶ 72]	**iErrorId:** The output issues an error code in the event of an error. *bError* goes TRUE at the same time.
str_Data_Rec	EIB_REC [▶ 76]	Connected with the send and receive function blocks.

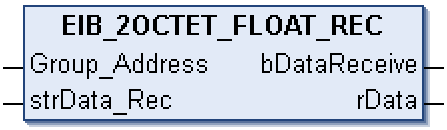
Requirements

Development environment	required TC3 PLC library
TwinCAT from v3.1.4022.29	Tc2_EIB from v3.3.9.0

4.2.4 Read

Function blocks	Description
EIB_2OCTET_FLOAT_REC [▸ 18]	Receiving of 2-byte Float EIB data and conversion to REAL
EIB_2OCTET_SIGN_REC [▸ 19]	Receiving of 2-byte Sign EIB data and conversion to INT
EIB_2OCTET_UNSIGN_REC [▸ 20]	Receiving of 2-byte Unsign EIB data and conversion to UINT
EIB_3BIT_CONTROL_REC [▸ 20]	Receiving of a "3 Bit Controlled" data type
EIB_4OCTET_FLOAT_REC [▸ 21]	Receiving of 4-byte Float EIB data and conversion to REAL
EIB_4OCTET_SIGN_REC [▸ 22]	Receiving of 4-byte Sign EIB data and conversion to DINT
EIB_4OCTET_UNSIGN_REC [▸ 22]	Receiving of 4-byte Unsign EIB data and conversion to UDINT
EIB_8BIT_SIGN_REC [▸ 23]	Receiving of 8 BIT EIB data and conversion to INT
EIB_8BIT_UNSIGN_REC [▸ 24]	Receiving of 8 BIT EIB data and conversion to BYTE
EIB_ALL_DATA_TYPES_REC [▸ 25]	Receives any EIB data
EIB_ALL_DATA_TYPES_REC_EX [▸ 25]	Receives any EIB data
EIB_BIT_CONTROL_REC [▸ 26]	Receiving of a "1 Bit Controlled" data type
EIB_BIT_REC [▸ 27]	Receiving of 1 BIT EIB data and conversion to BOOL
EIB_DATE_REC [▸ 28]	Receiving a date
EIB_TIME_REC [▸ 28]	Receiving a time

4.2.4.1 EIB_2OCTET_FLOAT_REC



This function block receives 2 bytes of float EIB data on the set group address and converts them into an IEC61131-3 REAL variable.

Inputs

```
VAR_INPUT
  Group_Address  : EIB_GROUP_ADDR;
  strData_Rec    : EIB_REC;
END_VAR
```

Name	Type	Description
Group_Address	EIB_GROUP_ADDR [▸ 74]	Group address from which the data were sent. This group address must be entered in the filters!
strData_Rec	EIB_REC [▸ 76]	The data structure with which the function block KL6301() [▸ 14] must be linked.

Outputs

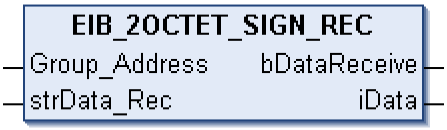
```
VAR_OUTPUT
  bDataReceive : BOOL;
  rData        : REAL;
END_VAR
```

Name	Type	Description
bDataReceive	BOOL	This bit is set to FALSE for exactly one cycle when an EIB telegram with the group address is received.
rData	REAL	Contains the user data of the received EIB telegram.

Requirements

Development environment	required TC3 PLC library
TwinCAT from v3.1.4020.14	Tc2_EIB from 3.3.4.0

4.2.4.2 EIB_2OCTET_SIGN_REC



This function block receives 2 bytes of sign EIB data on the set group address and converts them into an IEC61131-3 INT variable.

Inputs

```
VAR_INPUT
  Group_Address : EIB_GROUP_ADDR;
  strData_Rec   : EIB_REC;
END_VAR
```

Name	Type	Description
Group_Address	EIB_GROUP_ADDR [▶ 74]	Group address from which the data were sent. This group address must be entered in the filters!
strData_Rec	EIB_REC [▶ 76]	The data structure with which the function block KL6301() [▶ 14] must be linked.

Outputs

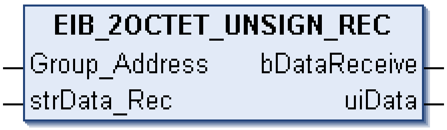
```
VAR_OUTPUT
  bDataReceive : BOOL;
  iData        : INT;
END_VAR
```

Name	Type	Description
bDataReceive	BOOL	This bit is set to FALSE for exactly one cycle when an EIB telegram with the group address is received.
iData	INT	Contains the user data of the received EIB telegram.

Requirements

Development environment	required TC3 PLC library
TwinCAT from v3.1.4020.14	Tc2_EIB from 3.3.4.0

4.2.4.3 EIB_2OCTET_UNSIGN_REC



This function block receives 2 bytes of unsign EIB data on the set group address and converts them into an IEC61131-3 UINT variable.

Inputs

```
VAR_INPUT
  Group_Address : EIB_GROUP_ADDR;
  strData_Rec   : EIB_REC;
END_VAR
```

Name	Type	Description
Group_Address	EIB_GROUP_ADDR [▶ 74]	Group address from which the data were sent. This group address must be entered in the filters!
strData_Rec	EIB_REC [▶ 76]	The data structure with which the function block KL6301() [▶ 14] must be linked.

Outputs

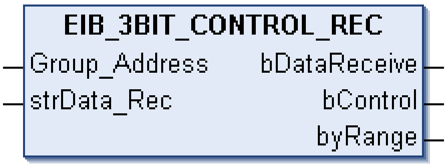
```
VAR_OUTPUT
  bDataReceive : BOOL;
  uiData       : UINT;
END_VAR
```

Name	Type	Description
bDataReceive	BOOL	This bit is set to FALSE for exactly one cycle when an EIB telegram with the group address is received.
uiData	UINT	Contains the user data of the received EIB telegram.

Requirements

Development environment	required TC3 PLC library
TwinCAT from v3.1.4020.14	Tc2_EIB from 3.3.4.0

4.2.4.4 EIB_3BIT_CONTROL_REC



This function block receives 4 bits of EIB data on the set group address and converts them into an IEC61131-3 BOOL variable and a BYTE variable.

Inputs

```
VAR_INPUT
  Group_Address : EIB_GROUP_ADDR;
  strData_Rec   : EIB_REC;
END_VAR
```

Name	Type	Description
Group_Address	EIB_GROUP_ADDR [▶ 74]	Group address from which the data were sent. This group address must be entered in the filters!
strData_Rec	EIB_REC [▶ 76]	The data structure with which the function block KL6301() [▶ 14] must be linked.

Outputs

```
VAR_OUTPUT
  bDataReceive : BOOL;
  bControl     : BOOL;
  byRange      : BYTE;
END_VAR
```

Name	Type	Description
bDataReceive	BOOL	This bit is set to FALSE for exactly one cycle when an EIB telegram with the group address is received.
byControl	BOOL	Valid values (TRUE/FALSE)
ByRange	BYTE	Valid values (000b..111b)

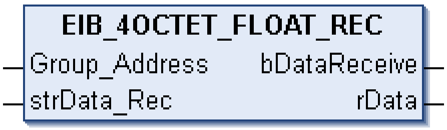
Allocation of the 4 bits to the variables bControl and byRange.

bControl	byRange.2	byRange.1	byRange.0

Requirements

Development environment	required TC3 PLC library
TwinCAT from v3.1.4020.14	Tc2_EIB from 3.3.4.0

4.2.4.5 EIB_4OCTET_FLOAT_REC



This function block receives 4 bytes of float EIB data on the set group address and converts them into an IEC61131-3 REAL variable.

Inputs

```
VAR_INPUT
  Group_Address : EIB_GROUP_ADDR;
  strData_Rec   : EIB_REC;
END_VAR
```

Name	Type	Description
Group_Address	EIB_GROUP_ADDR [▶ 74]	Group address from which the data were sent. This group address must be entered in the filters!
strData_Rec	EIB_REC [▶ 76]	The data structure with which the function block KL6301() [▶ 14] must be linked.

Outputs

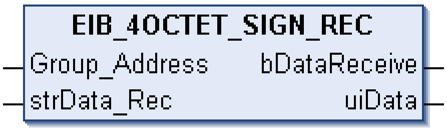
```
VAR_OUTPUT
  bDataReceive : BOOL;
  rData        : REAL;
END_VAR
```

Name	Type	Description
bDataReceive	BOOL	This bit is set to FALSE for exactly one cycle when an EIB telegram with the group address is received.
rData	REAL	Contains the user data of the received EIB telegram.

Requirements

Development environment	required TC3 PLC library
TwinCAT from v3.1.4020.14	Tc2_EIB from 3.3.4.0

4.2.4.6 EIB_4OCTET_SIGN_REC



This function block receives 4 bytes of sign EIB data on the set group address and converts them into an IEC61131-3 DINT variable.

Inputs

```
VAR_INPUT
  Group_Address : EIB_GROUP_ADDR;
  strData_Rec   : EIB_REC;
END_VAR
```

Name	Type	Description
Group_Address	EIB_GROUP_ADDR [▶ 74]	Group address from which the data were sent. This group address must be entered in the filters!
strData_Rec	EIB_REC [▶ 76]	The data structure with which the function block KL6301() [▶ 14] must be linked.

Outputs

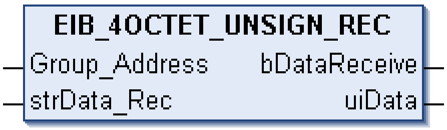
```
VAR_OUTPUT
  bDataReceive : BOOL;
  uiData       : DINT;
END_VAR
```

Name	Type	Description
bDataReceive	BOOL	This bit is set to FALSE for exactly one cycle when an EIB telegram with the group address is received.
uiData	DINT	Contains the user data of the received EIB telegram.

Requirements

Development environment	required TC3 PLC library
TwinCAT from v3.1.4020.14	Tc2_EIB from 3.3.4.0

4.2.4.7 EIB_4OCTET_UNSIGN_REC



This function block receives 4 bytes of unsign EIB data on the set group address and converts them into an IEC61131-3 UDINT variable.

Inputs

```
VAR_INPUT
  Group_Address : EIB_GROUP_ADDR;
  strData_Rec   : EIB_REC;
END_VAR
```

Name	Type	Description
Group_Address	EIB_GROUP_ADDR [▶ 74]	Group address from which the data were sent. This group address must be entered in the filters!
strData_Rec	EIB_REC [▶ 76]	The data structure with which the function block KL6301() [▶ 14] must be linked.

Outputs

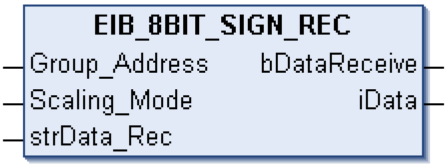
```
VAR_OUTPUT
  bDataReceive : BOOL;
  uiData       : UDINT;
END_VAR
```

Name	Type	Description
bDataReceive	BOOL	This bit is set to FALSE for exactly one cycle when an EIB telegram with the group address is received.
uiData	UDINT	Contains the user data of the received EIB telegram.

Requirements

Development environment	required TC3 PLC library
TwinCAT from v3.1.4020.14	Tc2_EIB from 3.3.4.0

4.2.4.8 EIB_8BIT_SIGN_REC



This function block receives 8 bits of EIB data on the set group address and converts them into an IEC61131-3 INT variable. In addition the value may be converted automatically.

Inputs

```
VAR_INPUT
  Group_Address : EIB_GROUP_ADDR;
  Scaling_Mode  : INT;
  strData_Rec   : EIB_REC;
END_VAR
```

Name	Type	Description
Group_Address	EIB_GROUP_ADDR [▶ 74]	Group address from which the data were sent. This group address must be entered in the filters!
Scaling_Mode	INT	0 - The 8-bit value is output as a % value 0...100% 1 - The 8-bit value is output as degrees (angle) 0...360° 2 - The 8-bit value is output as a byte value 0...255
strData_Rec	EIB_REC [▶ 76]	The data structure with which the function block KL6301() [▶ 14] must be linked.

Outputs

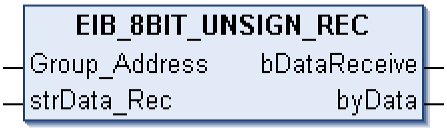
```
VAR_OUTPUT
  bDataReceive : BOOL;
  iData        : INT;
END_VAR
```

Name	Type	Description
bDataReceive	BOOL	This bit is set to FALSE for exactly one cycle when an EIB telegram with the group address is received.
iData	INT	Scaled value, see *Scaling_Mode* (-1, an invalid scaling mode was entered).

Requirements

Development environment	required TC3 PLC library
TwinCAT from v3.1.4020.14	Tc2_EIB from 3.3.4.0

4.2.4.9 EIB_8BIT_UNSIGN_REC



This function block receives 8 bits of EIB data on the set group address and converts them into an IEC61131-3 BYTE variable.

Inputs

```
VAR_INPUT
  Group_Address : EIB_GROUP_ADDR;
  strData_Rec   : EIB_REC;
END_VAR
```

Name	Type	Description
Group_Address	EIB_GROUP_ADDR [▶ 74]	Group address from which the data were sent. This group address must be entered in the filters!
strData_Rec	EIB_REC [▶ 76]	The data structure with which the function block KL6301() [▶ 14] must be linked.

Outputs

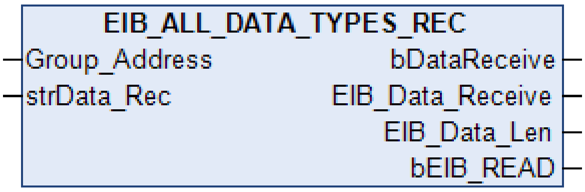
```
VAR_OUTPUT
  bDataReceive : BOOL;
  byData       : BYTE;
END_VAR
```

Name	Type	Description
bDataReceive	BOOL	This bit is set to FALSE for exactly one cycle when an EIB telegram with the group address is received.
byData	BYTE	Contains the user data of the received EIB telegram.

Requirements

Development environment	required TC3 PLC library
TwinCAT from v3.1.4020.14	Tc2_EIB from 3.3.4.0

4.2.4.10 EIB_ALL_DATA_TYPES_REC



This function block receives variable EIB data sizes on the set group address and outputs the raw data as a byte ARRAY.

Inputs

```
VAR_INPUT
  Group_Address : EIB_GROUP_ADDR;
  strData_Rec   : EIB_REC;
END_VAR
```

Name	Type	Description
Group_Address	EIB_GROUP_ADDR [▶ 74]	Group address from which the data were sent. This group address must be entered in the filters!
strData_Rec	EIB_REC [▶ 76]	The data structure with which the function block KL6301() [▶ 14] must be linked.

Outputs

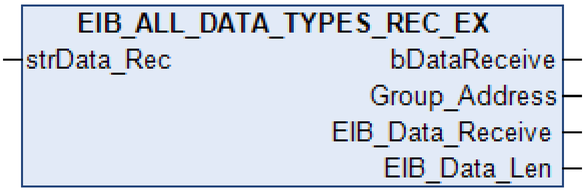
```
VAR_OUTPUT
  bDataReceive     : BOOL;
  EIB_Data_Receive : ARRAY [1..14] OF BYTE;
  EIB_Data_Len     : USINT;
  bEIB_READ        : BOOL;
END_VAR
```

Name	Type	Description
bDataReceive	BOOL	This bit is set to FALSE for exactly one cycle when an EIB telegram with the group address is received.
EIB_Data_Receive	ARRAY OF BYTE	Contains the user data of the received EIB telegram.
EIB_Data_Len	USINT	Contains the user data length of the receiving EIB telegram. Data < 8 bits specified length 1 Data >=) 8 bits specified length +1 Example: If you receive 1 bit of data, the length in EIB_Data_Len is 1. If you receive 2 bytes of data, the length in EIB_Data_Len is 3.
bEIB_READ	BOOL	TRUE = EIB read command. FALSE = normal EIB telegram (from v3.3.5.0).

Requirements

Development environment	required TC3 PLC library
TwinCAT from v3.1.4020.14	Tc2_EIB from 3.3.4.0

4.2.4.11 EIB_ALL_DATA_TYPES_REC_EX



This function block receives variable EIB data for all group addresses and outputs the raw data as a byte array.

Inputs

```
VAR_INPUT
  strData_Rec    : EIB_REC;
END_VAR
```

Name	Type	Description
strData_Rec	EIB_REC [▶ 76]	The data structure with which the function block KL6301() [▶ 14] must be linked.

Outputs

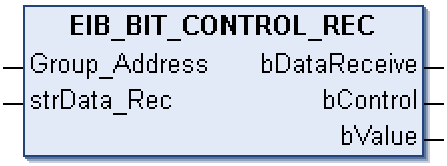
```
VAR_OUTPUT
  bDataReceive     : BOOL;
  Group_Address    : EIB_GROUP_ADDR;
  EIB_Data_Receive : ARRAY [1..14] OF BYTE;
  EIB_Data_Len     : USINT;
END_VAR
```

Name	Type	Description
bDataReceive	BOOL	This bit is set to FALSE for exactly one cycle when an EIB telegram with the group address is received.
Group_Address	EIB_GROUP_ADDR [▶ 74]	Group address from which the data were sent. This group address must be entered in the filters!
EIB_Data_Receive	ARRAY OF BYTE	Contains the user data of the received EIB telegram.
EIB_Data_Len	USINT	Contains the user data length of the receiving EIB telegram. / Data < 8 bits specified length 1, / Data >=) 8 bits specified length +1 / Example: If you receive 1 bit of data, the length in EIB_Data_Len is 1. If you receive 2 bytes of data, the length in EIB_Data_Len is 3.

Requirements

Development environment	required TC3 PLC library
TwinCAT from v3.1.4020.14	Tc2_EIB from 3.3.4.0

4.2.4.12 EIB_BIT_CONTROL_REC



This function block receives 2 bits of EIB data on the set group address and converts them into two IEC61131-3 BOOL variables.

Inputs

```
VAR_INPUT
  Group_Address : EIB_GROUP_ADDR;
  strData_Rec   : EIB_REC;
END_VAR
```

Name	Type	Description
Group_Address	EIB_GROUP_ADDR [▶ 74]	Group address from which the data were sent. This group address must be entered in the filters!
strData_Rec	EIB_REC [▶ 76]	The data structure with which the function block KL6301() [▶ 14] must be linked.

Outputs

```
VAR_OUTPUT
  bDataReceive : BOOL;
  bControl     : BOOL;
  bValue       : BOOL;
END_VAR
```

Name	Type	Description
bDataReceive	BOOL	This bit is set to FALSE for exactly one cycle when an EIB telegram with the group address is received.
bControl	BOOL	Valid values (TRUE/FALSE)
bValue	BOOL	Valid values (TRUE/FALSE)

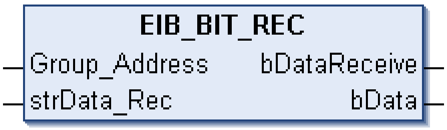
Allocation of the 2 bits to the variables bControl and bValue.

bControl	bValue

Requirements

Development environment	required TC3 PLC library
TwinCAT from v3.1.4020.14	Tc2_EIB from 3.3.4.0

4.2.4.13 EIB_BIT_REC



This function block receives 1 bit of EIB data on the set group address and converts them into an IEC61131-3 BOOL variable.

Inputs

```
VAR_INPUT
  Group_Address : EIB_GROUP_ADDR;
  strData_Rec   : EIB_REC;
END_VAR
```

Name	Type	Description
Group_Address	EIB_GROUP_ADDR [▶ 74]	Group address from which the data were sent. This group address must be entered in the filters!
strData_Rec	EIB_REC [▶ 76]	The data structure with which the function block KL6301() [▶ 14] must be linked.

Outputs

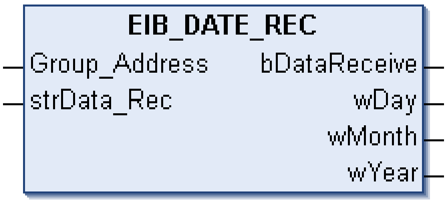
```
VAR_OUTPUT
  bDataReceive : BOOL;
  bData        : BOOL;
END_VAR
```

Name	Type	Description
bDataReceive	BOOL	This bit is set to FALSE for exactly one cycle when an EIB telegram with the group address is received.
bData	BOOL	Valid values (TRUE/FALSE)

Requirements

Development environment	required TC3 PLC library
TwinCAT from v3.1.4020.14	Tc2_EIB from 3.3.4.0

4.2.4.14 EIB_DATE_REC



This function block receives 3 bytes of EIB data on the set group address and converts them into three IEC61131-3 WORD variables.

Inputs

```
VAR_INPUT
  Group_Address : EIB_GROUP_ADDR;
  strData_Rec   : EIB_REC;
END_VAR
```

Name	Type	Description
Group_Address	EIB_GROUP_ADDR [▸ 74]	Group address from which the data were sent. This group address must be entered in the filters!
strData_Rec	EIB_REC [▸ 76]	The data structure with which the function block KL6301() [▸ 14] must be linked.

Outputs

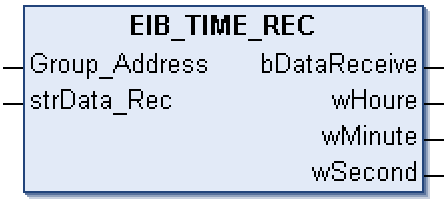
```
VAR_OUTPUT
  bDataReceive : BOOL;
  wDay         : WORD;
  wMonth       : WORD;
  wYear        : WORD;
END_VAR
```

Name	Type	Description
bDataReceive	BOOL	This bit is set to FALSE for exactly one cycle when an EIB telegram with the group address is received.
wDay	WORD	Date, days [1...31]
wMonth	WORD	Date, month [1...12]
wYear	WORD	Date, year [0...99]

Requirements

Development environment	required TC3 PLC library
TwinCAT from v3.1.4020.14	Tc2_EIB from 3.3.4.0

4.2.4.15 EIB_TIME_REC



This function block receives 3 bytes of EIB data on the set group address and converts them into three IEC61131-3 WORD variables.

Inputs

```
VAR_INPUT
  Group_Address : EIB_GROUP_ADDR;
  strData_Rec   : EIB_REC;
END_VAR
```

Name	Type	Description
Group_Address	EIB_GROUP_ADDR [▶ 74]	Group address from which the data were sent. This group address must be entered in the filters!
strData_Rec	EIB_REC [▶ 76]	The data structure with which the function block KL6301() [▶ 14] must be linked.

Outputs

```
VAR_OUTPUT
  bDataReceive : BOOL;
  wHoure       : WORD;
  wMinute      : WORD;
  wSecond      : WORD;
END_VAR
```

Name	Type	Description
bDataReceive	BOOL	This bit is set to FALSE for exactly one cycle when an EIB telegram with the group address is received.
wHoure	WORD	Time, in hours [0...23]
wMinute	WORD	Time, in minutes [0...59]
wSecond	WORD	Time, in seconds [0...59]

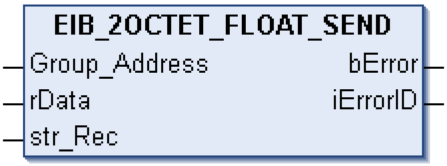
Requirements

Development environment	required TC3 PLC library
TwinCAT from v3.1.4020.14	Tc2_EIB from 3.3.4.0

4.2.5 Send

Function blocks	Description
EIB_2OCTET_FLOAT_SEND [▶ 30]	Sending a REAL value (conversion to 2 byte Float EIB)
EIB_2OCTET_FLOAT_SEND_EX [▶ 32]	Sending a REAL value (conversion to 2 byte Float EIB)
EIB_2OCTET_SIGN_SEND [▶ 33]	Sending an INT value (conversion to 2 byte Sign EIB)
EIB_2OCTET_SIGN_SEND_EX [▶ 34]	Sending an INT value (conversion to 2 byte Sign EIB)
EIB_2OCTET_UNSIGN_SEND [▶ 35]	Sending an UINT value (conversion to 2 byte Unsign EIB)
EIB_2OCTET_UNSIGN_SEND_EX [▶ 37]	Sending an UINT value (conversion to 2 byte Unsign EIB)
EIB_3BIT_CONTROL_SEND [▶ 38]	Sending a "3 bit Controlled" data type
EIB_3BIT_CONTROL_SEND_EX [▶ 39]	Sending a "3 bit Controlled" data type
EIB_4OCTET_FLOAT_SEND [▶ 40]	Sending a REAL value (conversion to 4 byte Float EIB)
EIB_4OCTET_FLOAT_SEND_EX [▶ 42]	Sending a REAL value (conversion to 4 byte Float EIB)
EIB_4OCTET_SIGN_SEND [▶ 43]	Sending a DINT value (conversion to 4 byte Sign EIB)
EIB_4OCTET_SIGN_SEND_EX [▶ 44]	Sending a DINT value (conversion to 4 byte Sign EIB)
EIB_4OCTET_UNSIGN_SEND [▶ 45]	Sending a UDINT value (conversion to 4 byte Unsign EIB)
EIB_4OCTET_UNSIGN_SEND_EX [▶ 47]	Sending a UDINT value (conversion to 4 byte Unsign EIB)
EIB_8BIT_SIGN_SEND [▶ 48]	Sending an INT value (conversion to 1 byte Sign EIB)
EIB_8BIT_SIGN_SEND_EX [▶ 49]	Sending an INT value (conversion to 1 byte Sign EIB)
EIB_8BIT_UNSIGN_SEND [▶ 51]	Sending a BYTE value (conversion to 1 byte Unsign EIB)
EIB_8BIT_UNSIGN_SEND_EX [▶ 52]	Sending a BYTE value (conversion to 1 byte Unsign EIB)
EIB_ALL_DATA_TYPES_SEND [▶ 54]	Sending any EIB data
EIB_BIT_CONTROL_SEND [▶ 56]	Sending a "1 bit Controlled" data type
EIB_BIT_CONTROL_SEND_EX [▶ 58]	Sending a "1 bit Controlled" data type
EIB_BIT_SEND [▶ 59]	Sending a BOOL value
EIB_BIT_SEND_EX [▶ 60]	Sending a BOOL value
EIB_BIT_SEND_MANUAL [▶ 61]	Sending a BOOL value
EIB_DATE_SEND [▶ 62]	Sending a date
EIB_DATE_SEND_EX [▶ 63]	Sending a date
EIB_READ_SEND [▶ 65]	Sending a *Read_Group_Req*
EIB_TIME_SEND [▶ 66]	Sending a time
EIB_TIME_SEND_EX [▶ 67]	Sending a time

4.2.5.1 EIB_2OCTET_FLOAT_SEND



This function block sends a 2-byte float EIB value to the set group address. An IEC61131-3 REAL value is available as input value. The data are only transferred if there is a change. If the value changes again within 1 second, new data are only sent to the EIB device after another second has passed (see diagram). No new EIB telegram is sent if the value changes within the "min. send time" but falls back to the old, already sent value within the "min. send time".

Inputs

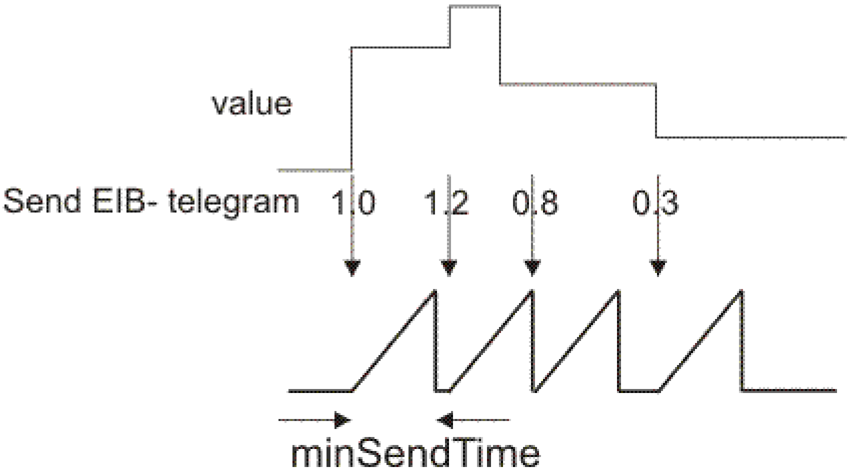
```
VAR_INPUT
  Group_Address : EIB_GROUP_ADDR;
  rData         : REAL;
  str_Rec       : EIB_REC;
END_VAR
```

Name	Type	Description
Group_Address	EIB_GROUP_ADDR [▸ 74]	Group address from which the data were sent. This group address must be entered in the filters!
rData	REAL	The data value in REAL. This is automatically converted to an EIB 2OCTET FLOAT value.
str_Rec	EIB_REC [▸ 76]	The data structure with which the function block KL6301() [▸ 14] must be linked.

Outputs

```
VAR_OUTPUT
  bError    : BOOL;
  iErrorID : EIB_ERROR_CODE;
END_VAR
```

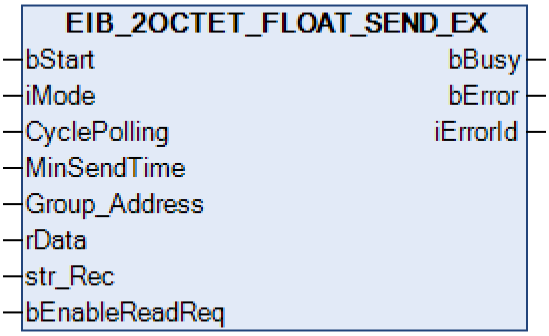
Name	Type	Description
bError	BOOL	This output goes TRUE as soon as an error occurs. This error is described via the *iErrorID* variable.
iErrorID	EIB_ERROR_CODE [▸ 72]	This output outputs an error code in the event of an error. *bError* goes TRUE at the same time.



Requirements

Development environment	required TC3 PLC library
TwinCAT from v3.1.4020.14	Tc2_EIB from 3.3.4.0

4.2.5.2 EIB_2OCTET_FLOAT_SEND_EX



This function block sends a 2-byte float EIB value to the set group address. An IEC61131-3 REAL value is available as input value. The data can be sent in Manual, Polling or OnChange mode (*iMode*).

Inputs

```
VAR_INPUT
  bStart         : BOOL;
  iMode          : INT;
  CyclePolling   : TIME := t#500ms;
  MinSendTime    : TIME := t#1s;
  Group_Address  : EIB_GROUP_ADDR;
  rData          : REAL;
  str_Rec        : EIB_REC;
  bEnableReadReq : BOOL;
END_VAR
```

Name	Type	Description
bStart	BOOL	The function block is activated by a positive edge at this input.
iMode	INT	0 - With a positive edge at *bStart*, an EIB telegram is sent. If the output *bBusy* is FALSE again, the command is processed. 1 - Polling mode: If *bStart* is TRUE, EIB telegrams are sent at intervals of *CyclePolling*. 2 - OnChange mode: If *bStart* is TRUE, an EIB telegram is automatically sent when the data changes. *MinSendTime* can be used to parameterize the minimum interval between two EIB messages, in order to avoid excessive EIB network load. 3 - OnChangePolling mode: If *bStart* is TRUE, EIB telegrams are sent at intervals of *CyclePolling* or automatically when the data changes. The minimum interval between two EIB messages is set with *MinSendTime*.
CyclePolling	TIME	Polling time for *iMode* = 1 (polling mode). The minimum time is 200 ms.
MinSendTime	TIME	Minimum interval time, which has to elapse before a telegram is sent in OnChange mode. The minimum time is 200 ms.
Group_Address	EIB_GROUP_ADDR [▶ 74]	Group address from which the data were sent. This group address must be entered in the filters!
rData	REAL	Contains the user data of the received EIB telegram.
str_Rec	EIB_REC [▶ 76]	The data structure with which the function block KL6301() [▶ 14] must be linked.
bEnableReadReq	BOOL	Enables the execution of read commands.

Outputs

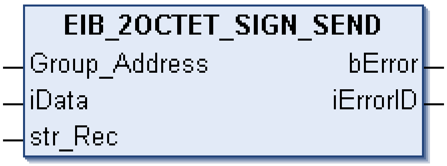
```
VAR_OUTPUT
  bBusy    : BOOL;
  bError   : BOOL;
  iErrorID : EIB_ERROR_CODE;
END_VAR
```

Name	Type	Description
bBusy	BOOL	The function block is active. Wait for new functions, until *bBusy* is FALSE again.
bError	BOOL	This output goes TRUE as soon as an error occurs. This error is described via the *iErrorID* variable.
iErrorID	EIB_ERROR_CODE [▶ 72]	This output outputs an error code in the event of an error. *bError* goes TRUE at the same time.

Requirements

Development environment	required TC3 PLC library
TwinCAT from v3.1.4020.14	Tc2_EIB from 3.3.4.0

4.2.5.3 EIB_2OCTET_SIGN_SEND



This function block sends a 2-byte sign EIB value to the set group address. An IEC61131-3 INT value is available as input value. The data are only transferred if there is a change. If the value changes again within 1 second, new data are only sent to the EIB device after another second has passed (see diagram). No new EIB telegram is sent if the value changes within the "min. send time" but falls back to the old, already sent value within the "min. send time".

Inputs

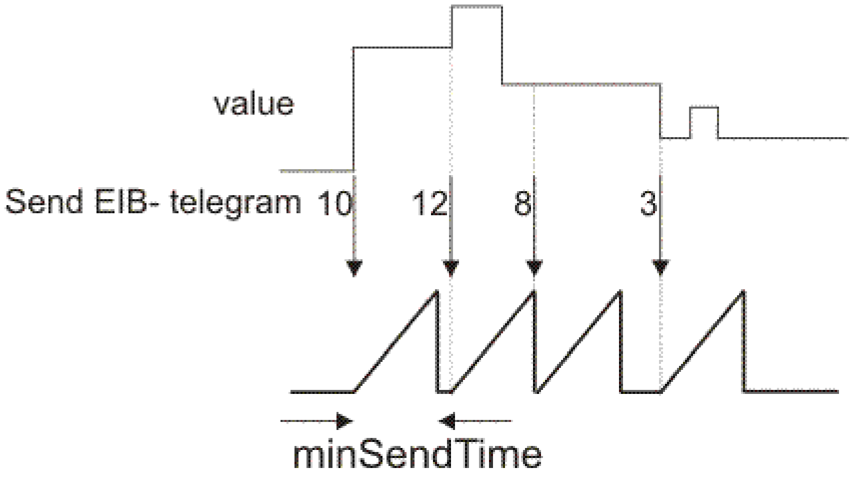
```
VAR_INPUT
  Group_Address : EIB_GROUP_ADDR;
  iData         : INT;
  str_Rec       : EIB_REC;
END_VAR
```

Name	Type	Description
Group_Address	EIB_GROUP_ADDR [▶ 74]	Group address to which the data are sent. This group address must be entered in the filters!
iData	INT	The data value in INT is automatically converted to an EIB 2OCTET SIGN value.
str_Rec	EIB_REC [▶ 76]	The data structure with which the function block KL6301() [▶ 14] must be linked.

Outputs

```
VAR_OUTPUT
  bError   : BOOL;
  iErrorID : EIB_ERROR_CODE;
END_VAR
```

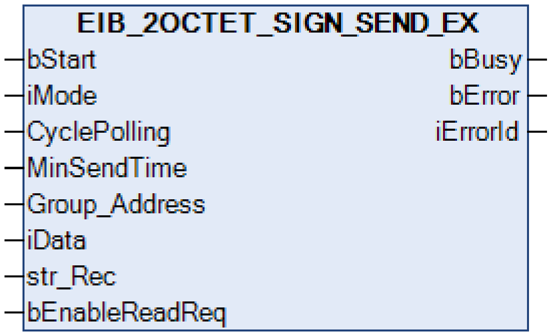
Name	Type	Description
bError	BOOL	This output goes TRUE as soon as an error occurs. This error is described via the *iErrorID* variable.
iErrorID	EIB_ERROR_CODE [▶ 72]	This output outputs an error code in the event of an error. *bError* goes TRUE at the same time.



Requirements

Development environment	required TC3 PLC library
TwinCAT from v3.1.4020.14	Tc2_EIB from 3.3.4.0

4.2.5.4 EIB_2OCTET_SIGN_SEND_EX



This function block sends a 2-byte sign EIB value to the set group address. An IEC61131-3 INT value is available as the input value. The data can be sent in Manual, Polling or OnChange depending on the set mode (*iMode*).

Inputs

```
VAR_INPUT
  bStart         : BOOL;
  iMode          : INT;
  CyclePolling   : TIME := t#500ms;
  MinSendTime    : TIME := t#1s;
  Group_Address  : EIB_GROUP_ADDR;
  iData          : INT;
  str_Rec        : EIB_REC;
  bEnableReadReq : BOOL;
END_VAR
```

Name	Type	Description
bStart	BOOL	Activates the function block. The function block starts operating depending on the set mode (see *iMode*).
iMode	INT	0 - With a positive edge at *bStart*, an EIB telegram is sent. If the output *bBusy* is FALSE again, the command is processed. 1 - Polling mode: If *bStart* is TRUE, EIB telegrams are sent at intervals of *CyclePolling*. 2 - OnChange mode: If *bStart* is TRUE, an EIB telegram is automatically sent when the data changes. *MinSendTime* can be used to parameterize the minimum interval between two EIB messages, in order to avoid excessive EIB network load. 3 - OnChangePolling mode: If *bStart* is TRUE, EIB telegrams are sent at intervals of *CyclePolling* or automatically when the data changes. The minimum interval between two EIB messages is set with *MinSendTime*.
CyclePolling	TIME	Polling time for *iMode* = 1 (polling mode). The minimum time is 200 ms.
MinSendTime	TIME	Minimum interval time, which has to elapse before a telegram is sent in OnChange mode. The minimum time is 200 ms.
Group_Address	EIB_GROUP_ADDR [▶ 74]	Group address to which the data are sent. This group address must be entered in the filters!
iData	INT	The data value in INT is automatically converted to an EIB 2OCTET SIGN value.
str_Rec	EIB_REC [▶ 76]	The data structure with which the function block KL6301() [▶ 14] must be linked.
bEnableReadReq	BOOL	Enables the execution of read commands.

Outputs

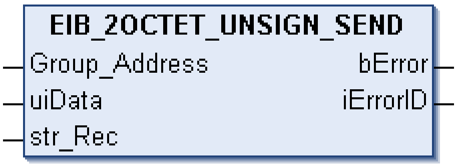
```
VAR_OUTPUT
  bBusy    : BOOL;
  bError   : BOOL;
  iErrorID : EIB_ERROR_CODE;
END_VAR
```

Name	Type	Description
bBusy	BOOL	The function block is active. Wait for new functions, until *bBusy* is FALSE again.
bError	BOOL	This output goes TRUE as soon as an error occurs. This error is described via the *iErrorID* variable.
iErrorID	EIB_ERROR_CODE [▶ 72]	This output outputs an error code in the event of an error. *bError* goes TRUE at the same time.

Requirements

Development environment	required TC3 PLC library
TwinCAT from v3.1.4022.29	Tc2_EIB from v3.3.9.0

4.2.5.5 EIB_2OCTET_UNSIGN_SEND



This function block sends a 2-byte unsign EIB value to the set group address. An IEC61131-3 UINT value is available as input value. The data are only transferred if there is a change. If the value changes again within 1 second, new data are only sent to the EIB device after another second has passed (see diagram). No new EIB telegram is sent if the value changes within the "min. send time" but falls back to the old, already sent value within the "min. send time".

Inputs

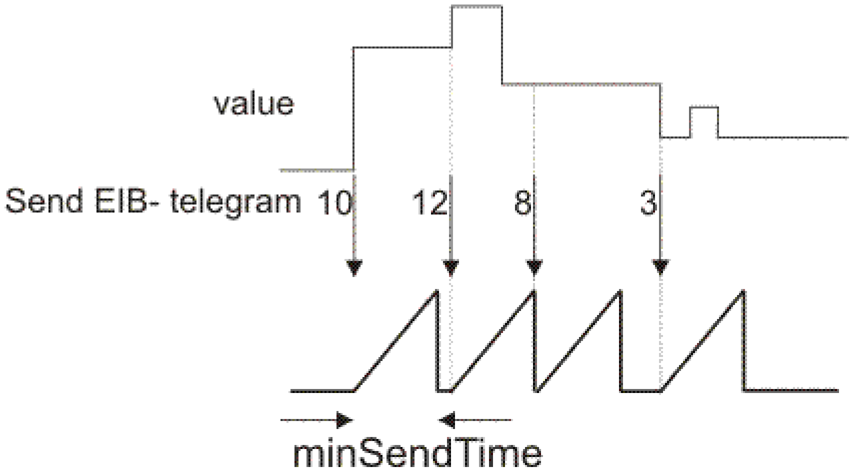
```
VAR_INPUT
  Group_Address : EIB_GROUP_ADDR;
  uiData        : UINT;
  str_Rec       : EIB_REC;
END_VAR
```

Name	Type	Description
Group_Address	EIB_GROUP_ADDR [▶ 74]	Group address to which the data are sent. This group address must be entered in the filters!
uiData	UINT	The data value in UINT is automatically converted to an EIB 2OCTET UNSIGN value.
str_Rec	EIB_REC [▶ 76]	The data structure with which the function block KL6301() [▶ 14] must be linked.

Outputs

```
VAR_OUTPUT
  bError    : BOOL;
  iErrorID : EIB_ERROR_CODE;
END_VAR
```

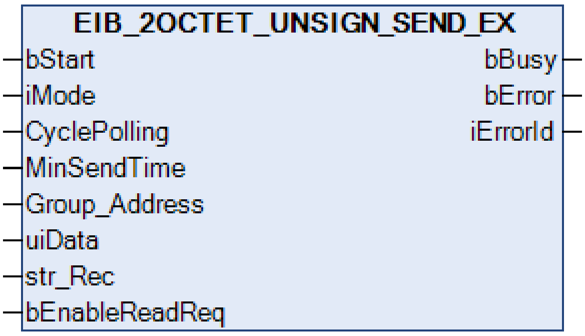
Name	Type	Description
bError	BOOL	This output goes TRUE as soon as an error occurs. This error is described via the *iErrorID* variable.
iErrorID	EIB_ERROR_CODE [▶ 72]	This output outputs an error code in the event of an error. *bError* goes TRUE at the same time.



Requirements

Development environment	required TC3 PLC library
TwinCAT from v3.1.4020.14	Tc2_EIB from 3.3.4.0

4.2.5.6 EIB_2OCTET_UNSIGN_SEND_EX



This function block sends a 2-byte Unsign EIB value to the set group address. An IEC61131-3 UINT value is available as input value. The data can be sent in Manual, Polling or OnChange depending on the set mode (*iMode*).

Inputs

```
VAR_INPUT
  bStart          : BOOL;
  iMode           : INT;
  CyclePolling    : TIME := t#500ms;
  MinSendTime     : TIME := t#1s;
  Group_Address   : EIB_GROUP_ADDR;
  uiData          : UINT;
  str_Rec         : EIB_REC;
  bEnableReadReq  : BOOL;
END_VAR
```

Name	Type	Description
bStart	BOOL	Activates the function block. The function block starts operating depending on the set mode (see *iMode*).
iMode	INT	0 - With a positive edge at *bStart*, an EIB telegram is sent. If the output *bBusy* is FALSE again, the command is processed. 1 - Polling mode: If *bStart* is TRUE, EIB telegrams are sent at intervals of *CyclePolling*. 2 - OnChange mode: If *bStart* is TRUE, an EIB telegram is automatically sent when the data changes. *MinSendTime* can be used to parameterize the minimum interval between two EIB messages, in order to avoid excessive EIB network load. 3 - OnChangePolling mode: If *bStart* is TRUE, EIB telegrams are sent at intervals of *CyclePolling* or automatically when the data changes. The minimum interval between two EIB messages is set with *MinSendTime*.
CyclePolling	TIME	Polling time for *iMode* = 1 (polling mode). The minimum time is 200 ms.
MinSendTime	TIME	Minimum interval time, which has to elapse before a telegram is sent in OnChange mode. The minimum time is 200 ms.
Group_Address	EIB_GROUP_ADDR [▶ 74]	Group address to which the data are sent. This group address must be entered in the filters!
uiData	UINT	The data value in UINT is automatically converted to an EIB 2OCTET UNSIGN value.
str_Rec	EIB_REC [▶ 76]	The data structure with which the function block KL6301() [▶ 14] must be linked.
bEnableReadReq	BOOL	Enables the execution of read commands.

Outputs

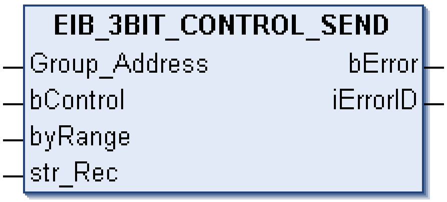
```
VAR_OUTPUT
  bBusy    : BOOL;
  bError   : BOOL;
  iErrorID : EIB_ERROR_CODE;
END_VAR
```

Name	Type	Description
bBusy	BOOL	The function block is active. Wait for new functions, until *bBusy* is FALSE again.
bError	BOOL	This output goes TRUE as soon as an error occurs. This error is described via the *iErrorID* variable.
iErrorID	EIB_ERROR_CODE [▶ 72]	This output outputs an error code in the event of an error. *bError* goes TRUE at the same time.

Requirements

Development environment	required TC3 PLC library
TwinCAT from v3.1.4024.52	Tc2_EIB from v3.4.2.0

4.2.5.7 EIB_3BIT_CONTROL_SEND



This function block sends a 4-bit EIB value to the set group address. An IEC61131-3 BOOL and a BYTE value are available as input value. The data are only transferred if there is a change in one of the two data types. If the value changes again within 200 milliseconds, new data are only sent to the EIB device after another 200 millisecond has passed (see diagram). No new EIB telegram is sent if the value changes within the "min. send time" but falls back to the old, already sent value within the "min. send time".

Inputs

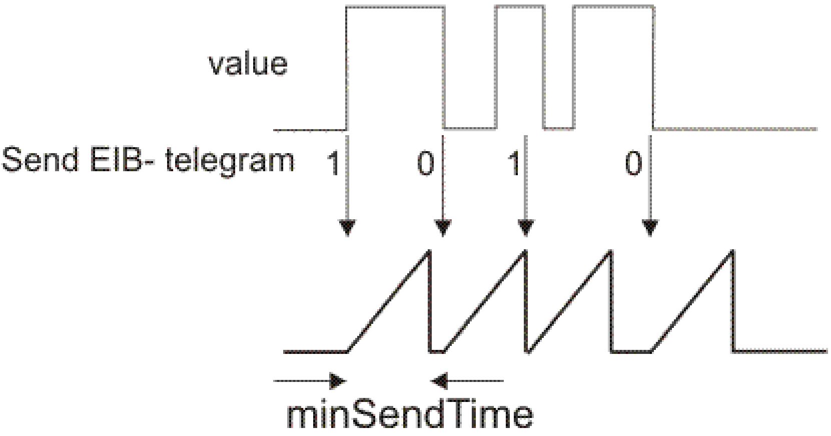
```
VAR_INPUT
  Group_Address : EIB_GROUP_ADDR;
  bControl      : BOOL;
  byRange       : BYTE;
  str_Rec       : EIB_REC;
END_VAR
```

Name	Type	Description
Group_Address	EIB_GROUP_ADDR [▶ 74]	Group address to which the data are sent. This group address must be entered in the filters!
bControl	BOOL	Value range TRUE/FALSE
byRange	BYTE	Value range 000b..111b
str_Rec	EIB_REC [▶ 76]	The data structure with which the function block KL6301() [▶ 14] must be linked.

Outputs

```
VAR_OUTPUT
  bError   : BOOL;
  iErrorID : EIB_ERROR_CODE;
END_VAR
```

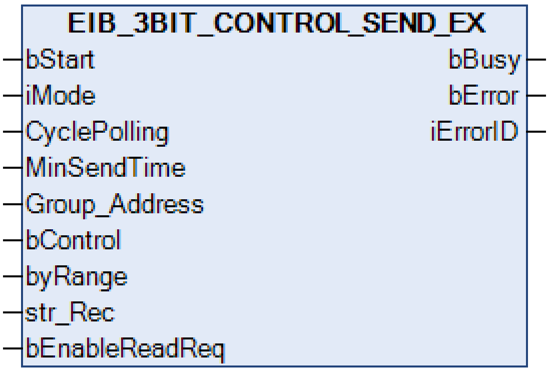
Name	Type	Description
bError	BOOL	This output goes TRUE as soon as an error occurs. This error is described via the *iErrorID* variable.
iErrorID	EIB_ERROR_CODE [▶ 72]	This output outputs an error code in the event of an error. *bError* goes TRUE at the same time.



Requirements

Development environment	required TC3 PLC library
TwinCAT from v3.1.4020.14	Tc2_EIB from 3.3.4.0

4.2.5.8 EIB_3BIT_CONTROL_SEND_EX



This function block sends a 4-bit EIB value to the set group address. An IEC61131-3 BOOL and a BYTE value are available as input value. The data can be sent in Manual, Polling or OnChange depending on the set mode (*iMode*).

Inputs

```
VAR_INPUT
  bStart         : BOOL;
  iMode          : INT;
  CyclePolling   : TIME := t#500ms;
  MinSendTime    : TIME := t#1s;
  Group_Address  : EIB_GROUP_ADDR;
  bControl       : BOOL;
  byRange        : BYTE;
  str_Rec        : EIB_REC;
  bEnableReadReq : BOOL;
END_VAR
```

Name	Type	Description
bStart	BOOL	Activates the function block. The function block starts operating depending on the set mode (see *iMode*).
iMode	INT	0 - With a positive edge at *bStart*, an EIB telegram is sent. If the output *bBusy* is FALSE again, the command is processed. 1 - Polling mode: If *bStart* is TRUE, EIB telegrams are sent at intervals of *CyclePolling*. 2 - OnChange mode: If *bStart* is TRUE, an EIB telegram is automatically sent when the data changes. *MinSendTime* can be used to parameterize the minimum interval between two EIB messages, in order to avoid excessive EIB network load. 3 - OnChangePolling mode: If *bStart* is TRUE, EIB telegrams are sent at intervals of *CyclePolling* or automatically when the data changes. The minimum interval between two EIB messages is set with *MinSendTime*.
CyclePolling	TIME	Polling time for *iMode* = 1 (polling mode). The minimum time is 200 ms.
MinSendTime	TIME	Minimum interval time, which has to elapse before a telegram is sent in OnChange mode. The minimum time is 200 ms.
Group_Address	EIB_GROUP_ADDR [▶ 74]	Group address to which the data are sent. This group address must be entered in the filters!
bControl	BOOL	Value range TRUE/FALSE
byRange	BYTE	Value range 000b..111b
str_Rec	EIB_REC [▶ 76]	The data structure with which the function block KL6301() [▶ 14] must be linked.
bEnableReadReq	BOOL	Enables the execution of read commands.

Outputs

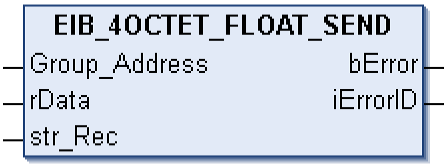
```
VAR_OUTPUT
  bBusy     : BOOL;
  bError    : BOOL;
  iErrorID  : EIB_ERROR_CODE;
END_VAR
```

Name	Type	Description
bBusy	BOOL	The function block is active. Wait for new functions, until *bBusy* is FALSE again.
bError	BOOL	This output goes TRUE as soon as an error occurs. This error is described via the *iErrorID* variable.
iErrorID	EIB_ERROR_CODE [▶ 72]	This output outputs an error code in the event of an error. *bError* goes TRUE at the same time.

Requirements

Development environment	required TC3 PLC library
TwinCAT from v3.1.4024.57	Tc2_EIB from v3.4.3.0

4.2.5.9 EIB_4OCTET_FLOAT_SEND



This function block sends a 4-byte float EIB value to the set group address. An IEC61131-3 REAL value is available as input value. The data are only transferred if there is a change. If the value changes again within 1 second, new data are only sent to the EIB device after another second has passed (see diagram). No new EIB telegram is sent if the value changes within the "min. send time" but falls back to the old, already sent value within the "min. send time".

Inputs

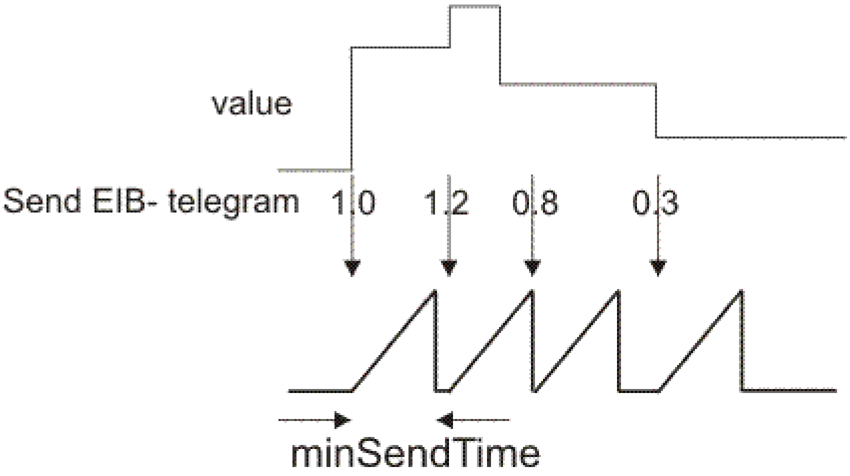
```
VAR_INPUT
  Group_Address : EIB_GROUP_ADDR;
  rData         : REAL;
  str_Rec       : EIB_REC;
END_VAR
```

Name	Type	Description
Group_Address	EIB_GROUP_ADDR [▶ 74]	Group address to which the data are sent. This group address must be entered in the filters!
rData	REAL	The data value in REAL. This is automatically converted to an EIB 2OCTET FLOAT value.
str_Rec	EIB_REC [▶ 76]	The data structure with which the function block KL6301() [▶ 14] must be linked.

Outputs

```
VAR_OUTPUT
  bError    : BOOL;
  iErrorID : EIB_ERROR_CODE;
END_VAR
```

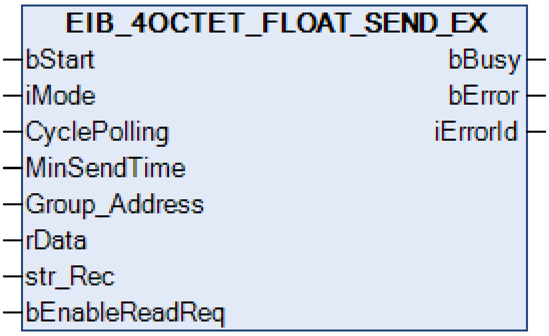
Name	Type	Description
bError	BOOL	This output goes TRUE as soon as an error occurs. This error is described via the *iErrorID* variable.
iErrorID	EIB_ERROR_CODE [▶ 72]	This output outputs an error code in the event of an error. *bError* goes TRUE at the same time.



Requirements

Development environment	required TC3 PLC library
TwinCAT from v3.1.4020.14	Tc2_EIB from 3.3.4.0

4.2.5.10 EIB_4OCTET_FLOAT_SEND_EX



This function block sends a 4-byte float EIB value to the set group address. An IEC61131-3 REAL value is available as input value. The data can be sent in Manual, Polling or OnChange mode (*iMode*).

Inputs

```
VAR_INPUT
  bStart          : BOOL;
  iMode           : INT;
  CyclePolling    : TIME := t#10m;
  MinSendTime     : TIME := t#1s;
  Group_Address   : EIB_GROUP_ADDR;
  rData           : REAL;
  str_Rec         : EIB_REC;
  bEnableReadReq  : BOOL;
END_VAR
```

Name	Type	Description
bStart	BOOL	Activates the function block. The function block starts operating depending on the set mode (see *iMode*).
iMode	INT	0 - With a positive edge at *bStart*, an EIB telegram is sent. If the output *bBusy* is FALSE again, the command is processed. 1 - Polling mode: If *bStart* is TRUE, EIB telegrams are sent at intervals of *CyclePolling*. 2 - OnChange mode: If *bStart* is TRUE, an EIB telegram is automatically sent when the data changes. *MinSendTime* can be used to parameterize the minimum interval between two EIB messages, in order to avoid excessive EIB network load. 3 - OnChangePolling mode: If *bStart* is TRUE, EIB telegrams are sent at intervals of *CyclePolling* or automatically when the data changes. The minimum interval between two EIB messages is set with *MinSendTime*.
CyclePolling	TIME	Polling time for *iMode* = 1 (polling mode). The minimum time is 200 ms.
MinSendTime	TIME	Minimum interval time, which has to elapse before a telegram is sent in OnChange mode. The minimum time is 200 ms.
Group_Address	EIB_GROUP_ADDR [▶ 74]	Group address to which the data are sent. This group address must be entered in the filters!
rData	REAL	Contains the user data of the received EIB telegram.
str_Rec	EIB_REC [▶ 76]	The data structure with which the function block KL6301() [▶ 14] must be linked.
bEnableReadReq	BOOL	Enables the execution of read commands.

Outputs

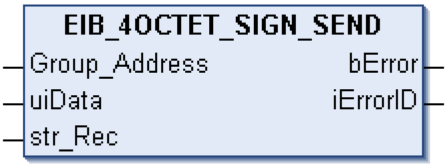
```
VAR_OUTPUT
  bBusy    : BOOL;
  bError   : BOOL;
  iErrorID : EIB_ERROR_CODE;
END_VAR
```

Name	Type	Description
bBusy	BOOL	The function block is active. Wait for new functions, until *bBusy* is FALSE again.
bError	BOOL	This output goes TRUE as soon as an error occurs. This error is described via the *iErrorID* variable.
iErrorID	EIB_ERROR_CODE [▶ 72]	This output outputs an error code in the event of an error. *bError* goes TRUE at the same time.

Requirements

Development environment	required TC3 PLC library
TwinCAT from v3.1.4020.14	Tc2_EIB from 3.3.4.0

4.2.5.11 EIB_4OCTET_SIGN_SEND



This function block sends a 4-byte sign EIB value to the set group address. An IEC61131-3 DINT value is available as input value. The data are only transferred if there is a change. If the value changes again within 1 second, new data are only sent to the EIB device after another second has passed (see diagram). No new EIB telegram is sent if the value changes within the "min. send time" but falls back to the old, already sent value within the "min. send time".

Inputs

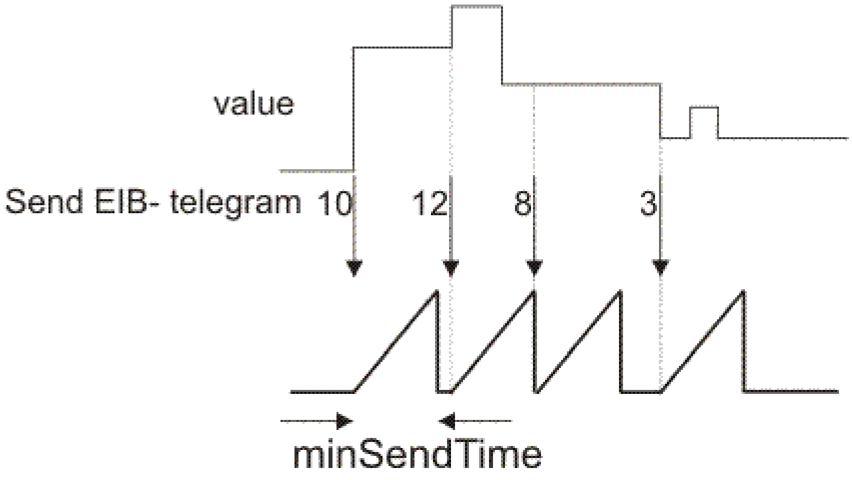
```
VAR_INPUT
  Group_Address : EIB_GROUP_ADDR;
  iData         : DINT;
  str_Rec       : EIB_REC;
END_VAR
```

Name	Type	Description
Group_Address	EIB_GROUP_ADDR [▶ 74]	Group address to which the data are sent. This group address must be entered in the filters!
uiData	DINT	The data value in DINT is automatically converted to an EIB 4OCTET SIGN value.
str_Rec	EIB_REC [▶ 76]	The data structure with which the function block KL6301() [▶ 14] must be linked.

Outputs

```
VAR_OUTPUT
  bError   : BOOL;
  iErrorID : EIB_ERROR_CODE;
END_VAR
```

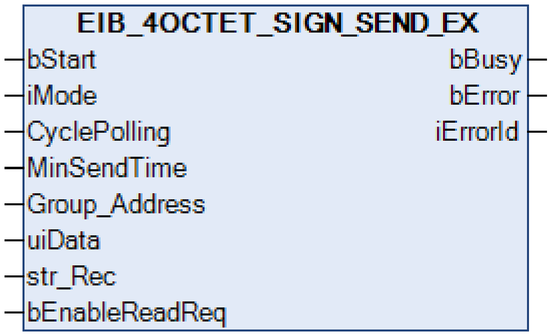
Name	Type	Description
bError	BOOL	This output goes TRUE as soon as an error occurs. This error is described via the *iErrorID* variable.
iErrorID	EIB_ERROR_CODE [▶ 72]	This output outputs an error code in the event of an error. *bError* goes TRUE at the same time.



Requirements

Development environment	required TC3 PLC library
TwinCAT from v3.1.4020.14	Tc2_EIB from 3.3.4.0

4.2.5.12 EIB_4OCTET_SIGN_SEND_EX



This function block sends a 4-byte sign EIB value to the set group address. An IEC61131-3 DINT value is available as input value. The data can be sent in Manual, Polling or OnChange mode (*iMode*).

Inputs

```
VAR_INPUT
  bStart         : BOOL;
  iMode          : INT;
  CyclePolling   : TIME := t#500ms;
  MinSendTime    : TIME := t#1s;
  Group_Address  : EIB_GROUP_ADDR;
  uiData         : DINT;
  str_Rec        : EIB_REC;
  bEnableReadReq : BOOL;
END_VAR
```

Name	Type	Description
bStart	BOOL	The function block is activated by a positive edge at this input.
iMode	INT	0 - With a positive edge at *bStart*, an EIB telegram is sent. If the output *bBusy* is FALSE again, the command is processed. 1 - Polling mode: If *bStart* is TRUE, EIB telegrams are sent at intervals of *CyclePolling*. 2 - OnChange mode: If *bStart* is TRUE, an EIB telegram is automatically sent when the data changes. *MinSendTime* can be used to parameterize the minimum interval between two EIB messages, in order to avoid excessive EIB network load. 3 - OnChangePolling mode: If *bStart* is TRUE, EIB telegrams are sent at intervals of *CyclePolling* or automatically when the data changes. The minimum interval between two EIB messages is set with *MinSendTime*.
CyclePolling	TIME	Polling time for *iMode* = 1 (polling mode). The minimum time is 200 ms.
MinSendTime	TIME	Minimum interval time, which has to elapse before a telegram is sent in OnChange mode. The minimum time is 200 ms.
Group_Address	EIB_GROUP_ADDR [▶ 74]	Group address to which the data are sent. This group address must be entered in the filters!
uiData	DINT	The data value in DINT is automatically converted to an EIB 4OCTET SIGN value.
str_Rec	EIB_REC [▶ 76]	The data structure with which the function block KL6301() [▶ 14] must be linked.
bEnableReadReq	BOOL	Enables the execution of read commands.

Outputs

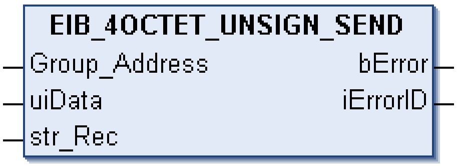
```
VAR_OUTPUT
  bBusy    : BOOL;
  bError   : BOOL;
  iErrorID : EIB_ERROR_CODE;
END_VAR
```

Name	Type	Description
bBusy	BOOL	The function block is active. Wait for new functions, until *bBusy* is FALSE again.
bError	BOOL	This output goes TRUE as soon as an error occurs. This error is described via the *iErrorID* variable.
iErrorID	EIB_ERROR_CODE [▶ 72]	This output outputs an error code in the event of an error. *bError* goes TRUE at the same time.

Requirements

Development environment	required TC3 PLC library
TwinCAT from v3.1.4020.14	Tc2_EIB from 3.3.4.0

4.2.5.13 EIB_4OCTET_UNSIGN_SEND



This function block sends a 4-byte unsign EIB value to the set group address. An IEC61131-3 UDINT value is available as input value. The data are only transferred if there is a change. If the value changes again within 1 second, new data are only sent to the EIB device after another second has passed (see diagram). No new EIB telegram is sent if the value changes within the "min. send time" but falls back to the old, already sent value within the "min. send time".

Inputs

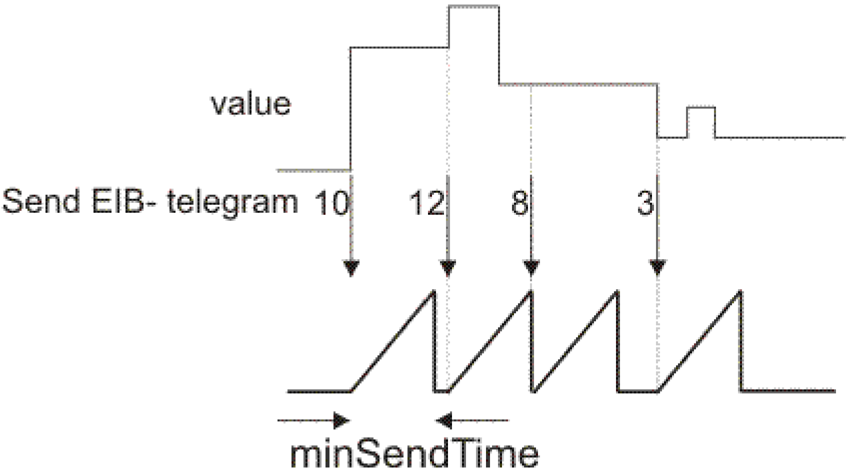
```
VAR_INPUT
  Group_Address   : EIB_GROUP_ADDR;
  uiData          : UDINT;
  str_Rec         : EIB_REC;
END_VAR
```

Name	Type	Description
Group_Address	EIB_GROUP_ADDR [▶ 74]	Group address to which the data are sent. This group address must be entered in the filters!
uiData	UDINT	The data value in UDINT is automatically converted to an EIB 4OCTET UNSIGN value.
str_Rec	EIB_REC [▶ 76]	The data structure with which the function block KL6301() [▶ 14] must be linked.

Outputs

```
VAR_OUTPUT
  bError    : BOOL;
  iErrorID : EIB_ERROR_CODE;
END_VAR
```

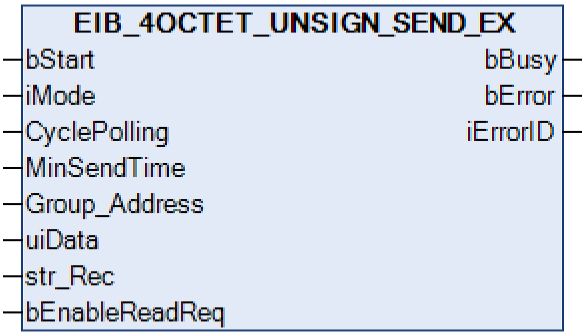
Name	Type	Description
bError	BOOL	This output goes TRUE as soon as an error occurs. This error is described via the *iErrorID* variable.
iErrorID	EIB_ERROR_CODE [▶ 72]	This output outputs an error code in the event of an error. *bError* goes TRUE at the same time.



Requirements

Development environment	required TC3 PLC library
TwinCAT from v3.1.4020.14	Tc2_EIB from 3.3.4.0

4.2.5.14 EIB_4OCTET_UNSIGN_SEND_EX



This function block sends a 4-byte Unsign EIB value to the set group address. An IEC61131-3 UDINT value is available as input value. The data can be sent in Manual, Polling or OnChange depending on the set mode (*iMode*).

Inputs

```
VAR_INPUT
  bStart          : BOOL;
  iMode           : INT;
  CyclePolling    : TIME := t#500ms;
  MinSendTime     : TIME := t#1s;
  Group_Address   : EIB_GROUP_ADDR;
  uiData          : UDINT;
  str_Rec         : EIB_REC;
  bEnableReadReq  : BOOL;
END_VAR
```

Name	Type	Description
bStart	BOOL	The function block is activated by a positive edge at this input.
iMode	INT	0 - With a positive edge at *bStart*, an EIB telegram is sent. If the output *bBusy* is FALSE again, the command is processed. 1 - Polling mode: If *bStart* is TRUE, EIB telegrams are sent at intervals of *CyclePolling*. 2 - OnChange mode: If *bStart* is TRUE, an EIB telegram is automatically sent when the data changes. *MinSendTime* can be used to parameterize the minimum interval between two EIB messages, in order to avoid excessive EIB network load. 3 - OnChangePolling mode: If *bStart* is TRUE, EIB telegrams are sent at intervals of *CyclePolling* or automatically when the data changes. The minimum interval between two EIB messages is set with *MinSendTime*.
CyclePolling	TIME	Polling time for *iMode* = 1 (polling mode). The minimum time is 200 ms.
MinSendTime	TIME	Minimum interval time, which has to elapse before a telegram is sent in OnChange mode. The minimum time is 200 ms.
Group_Address	EIB_GROUP_ADDR [▸ 74]	Group address to which the data are sent. This group address must be entered in the filters!
uiData	UDINT	The data value in UDINT is automatically converted to an EIB 4OCTET UNSIGN value.

str_Rec	EIB_REC [▸ 76]	The data structure with which the function block KL6301() [▸ 14] must be linked.
bEnableReadReq	BOOL	Enables the execution of read commands.

Outputs

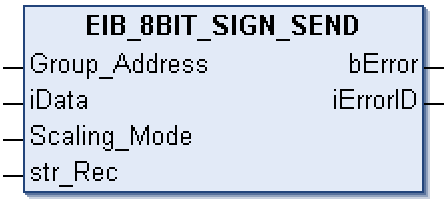
```
VAR_OUTPUT
  bBusy    : BOOL;
  bError   : BOOL;
  iErrorID : EIB_ERROR_CODE;
END_VAR
```

Name	Type	Description
bBusy	BOOL	The function block is active. Wait for new functions, until *bBusy* is FALSE again.
bError	BOOL	This output goes TRUE as soon as an error occurs. This error is described via the *iErrorID* variable.
iErrorID	EIB_ERROR_CODE [▸ 72]	This output outputs an error code in the event of an error. *bError* goes TRUE at the same time.

Requirements

Development environment	required TC3 PLC library
TwinCAT from v3.1.4024.57	Tc2_EIB from v3.4.3.0

4.2.5.15 EIB_8BIT_SIGN_SEND



This function block sends a 8-bit EIB value to the set group address. An IEC61131-3 INT value is available as input value. Scaling_Mode can be used to scale the input data value. The data are only transferred if there is a change in the data value. If the value changes again within 1 second, new data are only sent to the EIB device after "minSendTime" has passed (see diagram). No new EIB telegram is sent if the value changes within the "min. send time" but falls back to the old, already sent value within the "min. send time".

Inputs

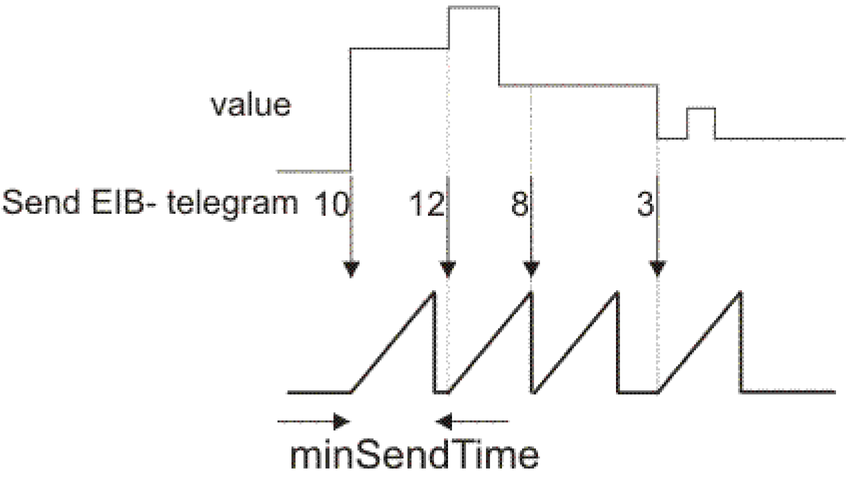
```
VAR_INPUT
  Group_Address   : EIB_GROUP_ADDR;
  iData           : INT;
  Scaling_Mode    : INT;
  str_Rec         : EIB_REC;
END_VAR
```

Name	Type	Description
Group_Address	EIB_GROUP_ADDR [▸ 74]	Group address to which the data are sent. This group address must be entered in the filters!
iData	INT	Data to be sent. Value range depending on *Scaling_Mode*.
Scaling_Mode	INT	0 - 0...100 [%] 1 - 0...360 [°] 2 - 0...255
str_Rec	EIB_REC [▸ 76]	The data structure with which the function block KL6301() [▸ 14] must be linked.

Outputs

```
VAR_OUTPUT
  bError    : BOOL;
  iErrorID : EIB_ERROR_CODE;
END_VAR
```

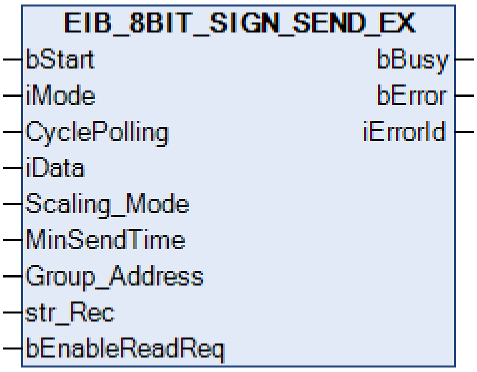
Name	Type	Description
bError	BOOL	This output goes TRUE as soon as an error occurs. This error is described via the *iErrorID* variable.
iErrorID	EIB_ERROR_CODE [▶ 72]	This output outputs an error code in the event of an error. *bError* goes TRUE at the same time.



Requirements

Development environment	required TC3 PLC library
TwinCAT from v3.1.4020.14	Tc2_EIB from 3.3.4.0

4.2.5.16 EIB_8BIT_SIGN_SEND_EX



This function block sends a 8-bit EIB value to the set group address. An IEC61131-3 INT value is available as input value. *Scaling_Mode* can be used to scale the input data value. The data can be sent in Manual, Polling or OnChange mode (*iMode*).

Inputs

```
VAR_INPUT
  bStart          : BOOL;
  iMode           : INT;
  CyclePolling    : TIME := t#500ms;
  iData           : INT;
  Scaling_Mode    : INT;
  MinSendTime     : TIME := t#1s;
  Group_Address   : EIB_GROUP_ADDR;
  str_Rec         : EIB_REC;
  bEnableReadReq  : BOOL;
END_VAR
```

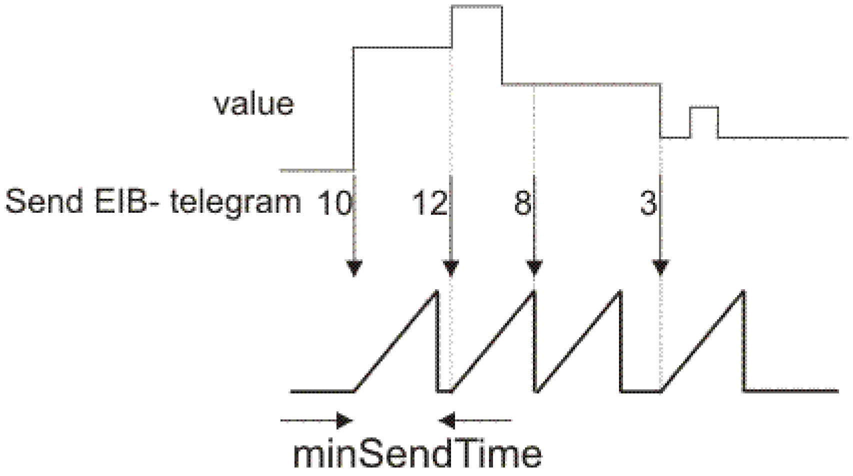
Name	Type	Description
bStart	BOOL	The function block is activated by a positive edge at this input.
iMode	INT	0 - With a positive edge at *bStart*, an EIB telegram is sent. If the output *bBusy* is FALSE again, the command is processed. 1 - Polling mode: If *bStart* is TRUE, EIB telegrams are sent at intervals of *CyclePolling*. 2 - OnChange mode: If *bStart* is TRUE, an EIB telegram is automatically sent when the data changes. *MinSendTime* can be used to parameterize the minimum interval between two EIB messages, in order to avoid excessive EIB network load. 3 - OnChangePolling mode: If *bStart* is TRUE, EIB telegrams are sent at intervals of *CyclePolling* or automatically when the data changes. The minimum interval between two EIB messages is set with *MinSendTime*.
CyclePolling	TIME	Polling time for *iMode* = 1 (polling mode). The minimum time is 200 ms.
iData	INT	Data to be sent. Value range depending on *Scaling_Mode*.
Scaling_Mode	INT	0 - 0...100 [%] 1 - 0...360 [°] 2 - 0...255
MinSendTime	TIME	Minimum interval time, which has to elapse before a telegram is sent in OnChange mode. The minimum time is 200 ms.
Group_Address	EIB_GROUP_ADDR [▶ 74]	Group address to which the data are sent. This group address must be entered in the filters!
str_Rec	EIB_REC [▶ 76]	The data structure with which the function block KL6301() [▶ 14] must be linked.
bEnableReadReq	BOOL	Enables the execution of read commands.

Outputs

```
VAR_OUTPUT
  bBusy    : BOOL;
  bError   : BOOL;
  iErrorID : EIB_ERROR_CODE;
END_VAR
```

Name	Type	Description
bBusy	BOOL	The function block is active. Wait for new functions, until *bBusy* is FALSE again.
bError	BOOL	This output goes TRUE as soon as an error occurs. This error is described via the *iErrorID* variable.
iErrorID	EIB_ERROR_CODE [▶ 72]	This output outputs an error code in the event of an error. *bError* goes TRUE at the same time.

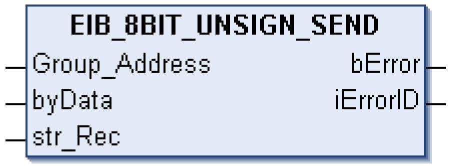
OnChange mode:



Requirements

Development environment	required TC3 PLC library
TwinCAT from v3.1.4020.14	Tc2_EIB from 3.3.4.0

4.2.5.17 EIB_8BIT_UNSIGN_SEND



This function block sends a 8-bit EIB value to the set group address. An IEC61131-3 byte value is available as input value. The data are only transferred if there is a change in the data value. If the value changes again within 1 second, new data are only sent to the EIB device after "minSendTime" has passed (see diagram). No new EIB telegram is sent if the value changes within the "min. send time" but falls back to the old, already sent value within the "min. send time".

Inputs

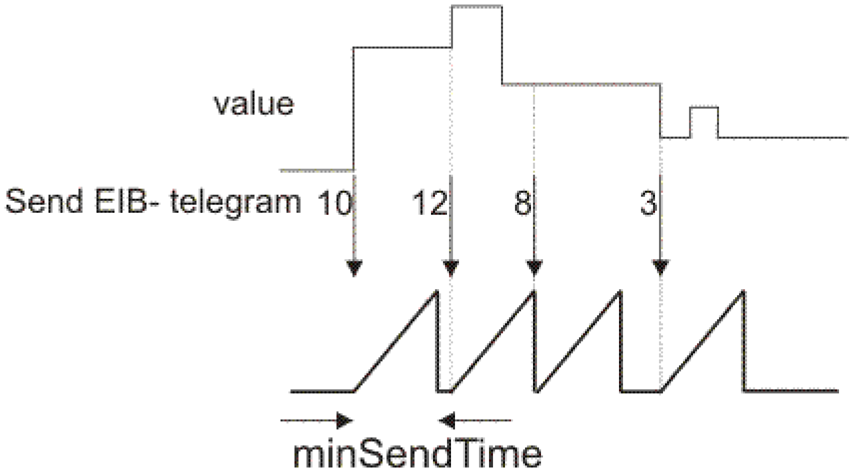
```
VAR_INPUT
  Group_Address   : EIB_GROUP_ADDR;
  byData          : BYTE;
  str_Rec         : EIB_REC;
END_VAR
```

Name	Type	Description
Group_Address	EIB_GROUP_ADDR [▸ 74]	Group address to which the data are sent. This group address must be entered in the filters!
byData	BYTE	Data to be sent. Value range 0x00...0xFF.
str_Rec	EIB_REC [▸ 76]	The data structure with which the function block KL6301() [▸ 14] must be linked.

Outputs

```
VAR_OUTPUT
  bError    : BOOL;
  iErrorID : EIB_ERROR_CODE;
END_VAR
```

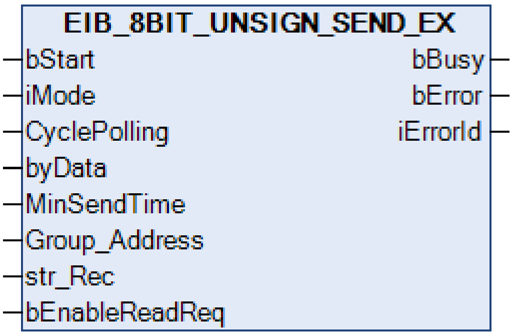
Name	Type	Description
bError	BOOL	This output goes TRUE as soon as an error occurs. This error is described via the *iErrorID* variable.
iErrorID	EIB_ERROR_CODE [▸ 72]	This output outputs an error code in the event of an error. *bError* goes TRUE at the same time.



Requirements

Development environment	required TC3 PLC library
TwinCAT from v3.1.4020.14	Tc2_EIB from 3.3.4.0

4.2.5.18 EIB_8BIT_UNSIGN_SEND_EX



This function block sends a 8-bit EIB value to the set group address. An IEC61131-3 byte value is available as input value. The data can be sent in Manual, Polling or OnChange mode (*iMode*).

Inputs

```
VAR_INPUT
  bStart         : BOOL;
  iMode          : INT;
  CyclePolling   : TIME := t#500ms;
  byData         : BYTE;
  MinSendTime    : TIME := t#1s;
  Group_Address  : EIB_GROUP_ADDR;
  str_Rec        : EIB_REC;
  bEnableReadReq : BOOL;
END_VAR
```

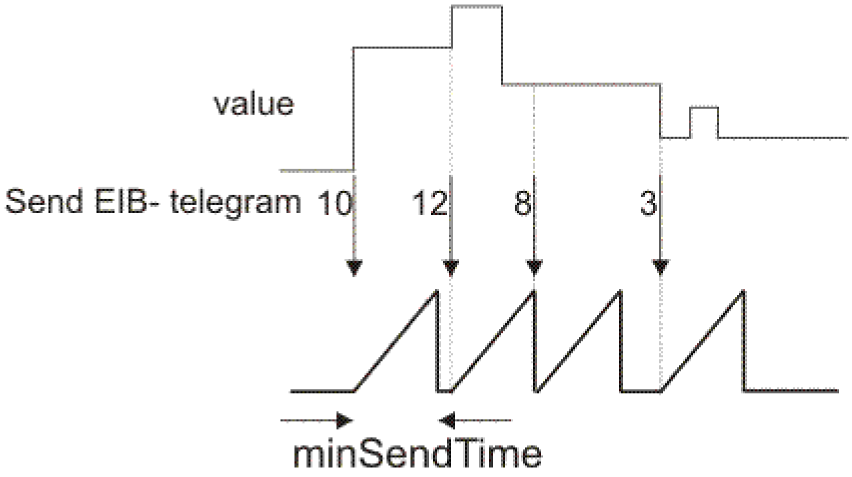
Name	Type	Description
bStart	BOOL	The function block is activated by a positive edge at this input.
iMode	INT	0 - With a positive edge at *bStart*, an EIB telegram is sent. If the output *bBusy* is FALSE again, the command is processed. 1 - Polling mode: If *bStart* is TRUE, EIB telegrams are sent at intervals of *CyclePolling*. 2 - OnChange mode: If *bStart* is TRUE, an EIB telegram is automatically sent when the data changes. *MinSendTime* can be used to parameterize the minimum interval between two EIB messages, in order to avoid excessive EIB network load. 3 - OnChangePolling mode: If *bStart* is TRUE, EIB telegrams are sent at intervals of *CyclePolling* or automatically when the data changes. The minimum interval between two EIB messages is set with *MinSendTime*.
CyclePolling	TIME	Polling time for *iMode* = 1 (polling mode). The minimum time is 200 ms.
byData	BYTE	Data to be sent. Value range 0x00...0xFF.
MinSendTime	TIME	Minimum interval time, which has to elapse before a telegram is sent in OnChange mode. The minimum time is 200 ms.
Group_Address	EIB_GROUP_ADDR [▶ 74]	Group address to which the data are sent. This group address must be entered in the filters!
str_Rec	EIB_REC [▶ 76]	The data structure with which the function block KL6301() [▶ 14] must be linked.
bEnableReadReq	BOOL	Enables the execution of read commands.

Outputs

```
VAR_OUTPUT
  bBusy    : BOOL;
  bError   : BOOL;
  iErrorID : EIB_ERROR_CODE;
END_VAR
```

Name	Type	Description
bBusy	BOOL	The function block is active. Wait for new functions, until *bBusy* is FALSE again.
bError	BOOL	This output goes TRUE as soon as an error occurs. This error is described via the *iErrorID* variable.
iErrorID	EIB_ERROR_CODE [▶ 72]	This output outputs an error code in the event of an error. *bError* goes TRUE at the same time.

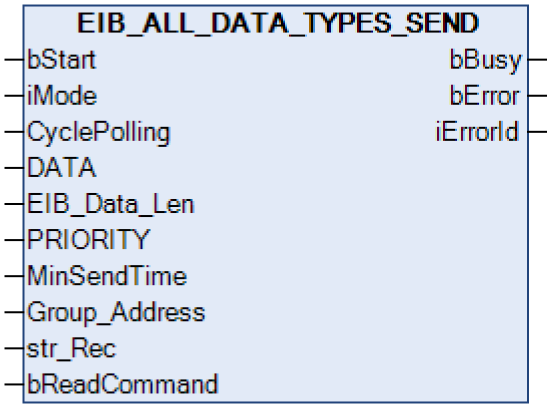
OnChange mode:



Requirements

Development environment	required TC3 PLC library
TwinCAT from v3.1.4020.14	Tc2_EIB from 3.3.4.0

4.2.5.19 EIB_ALL_DATA_TYPES_SEND



This function block sends a freely selectable EIB value to the set group address. An IEC61131-3 byte ARRAY variable is available as input value. The data are sent depending on the set mode.

Inputs

```
VAR_INPUT
  bStart        : BOOL;
  iMode         : INT;
  CyclePolling  : TIME := t#100ms;
  DATA          : ARRAY [1..14] OF BYTE;
  EIB_Data_Len  : USINT := 1;
  PRIORITY      : EIB_PRIORITY := EIB_PRIORITY_LOW;
  MinSendTime   : TIME := t#1s;
  Group_Address : EIB_GROUP_ADDR;
  str_Rec       : EIB_REC;
  bReadCommand  : BOOL;
END_VAR
```

Name	Type	Description
bStart	BOOL	If the mode is set to 0, an EIB telegram with positive edge is sent to *bStart*.
iMode	INT	0 - Manual (Fig. 1) 1 - Polling (Fig. 2) 2 - OnChange (Fig. 3)
CyclePolling	TIME	If mode 1 is selected, an EIB telegram is sent at the set time, even if the data values have not changed.
DATA	ARRAY OF BYTE	EIB data values.
EIB_Data_Len	USINT	Length of EIB data, EIB values >=) 1 byte calculate the length +1, EIB values < 1 byte set length equal to 1
PRIORITY	EIB_PRIORITY [▶ 74]	EIB priority, Low, High, Alarm.
MinSendTime	TIME	If mode 2 is selected, data are only transferred in the event of a change. *MinSendTime* defines the minimum time between two EIB telegrams.
Group_Address	EIB_GROUP_ADDR [▶ 74]	Group address to which the data are sent. This group address must be entered in the filters!
str_Rec	EIB_REC [▶ 76]	The data structure with which the function block KL6301() [▶ 14] must be linked.
bReadCommand	BOOL	A response can be sent to an EIB READ COMMAND.

Outputs

```
VAR_OUTPUT
  bBusy    : BOOL;
  bError   : BOOL;
  iErrorID : EIB_ERROR_CODE;
END_VAR
```

Name	Type	Description
bBusy	BOOL	The function block is active. Wait for new functions, until *bBusy* is FALSE again.
bError	BOOL	This output goes TRUE as soon as an error occurs. This error is described via the *iErrorID* variable.
iErrorID	EIB_ERROR_CODE [▶ 72]	This output outputs an error code in the event of an error. *bError* goes TRUE at the same time.

Transfer mode

Mode 0 manual

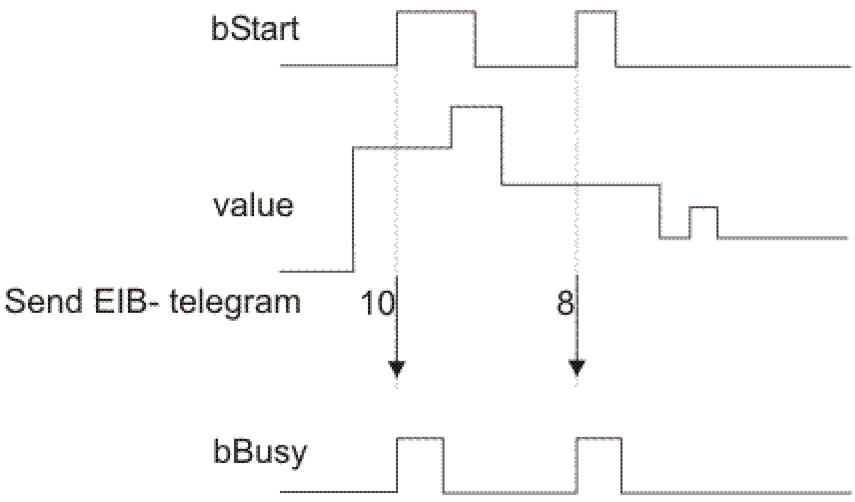


Figure 1

Mode 1 Polling

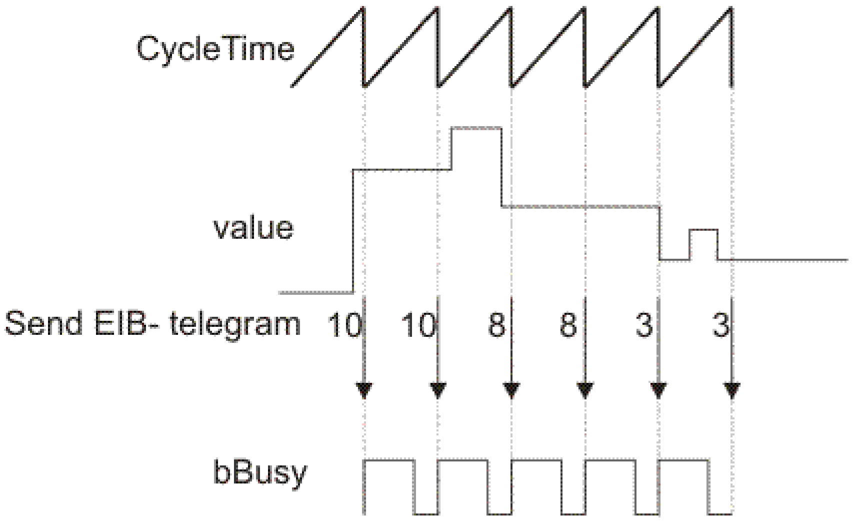


Figure 2

Mode 2 OnChange

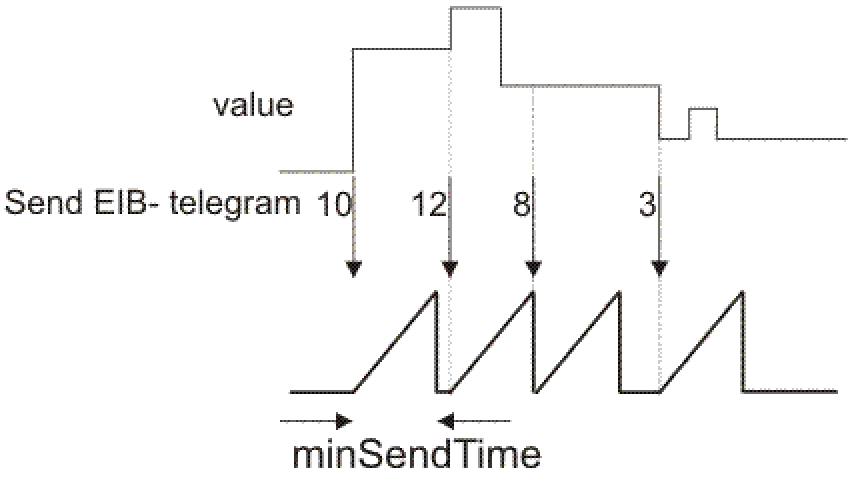
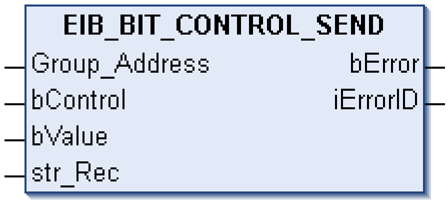


Abbildung 3

Requirements

Development environment	required TC3 PLC library
TwinCAT from v3.1.4020.14	Tc2_EIB from 3.3.4.0

4.2.5.20 EIB_BIT_CONTROL_SEND



This function block sends a 2-bit EIB value to the set group address. Two IEC61131-3 BOOL variables are available as input values. The data are only transferred if there is a change in one of the two data types. If the value changes again within 200 milliseconds, new data are only sent to the EIB device after another 200 millisecond has passed (see diagram). No new EIB telegram is sent if the value changes within the "min. send time" but falls back to the old, already sent value within the "min. send time".

Inputs

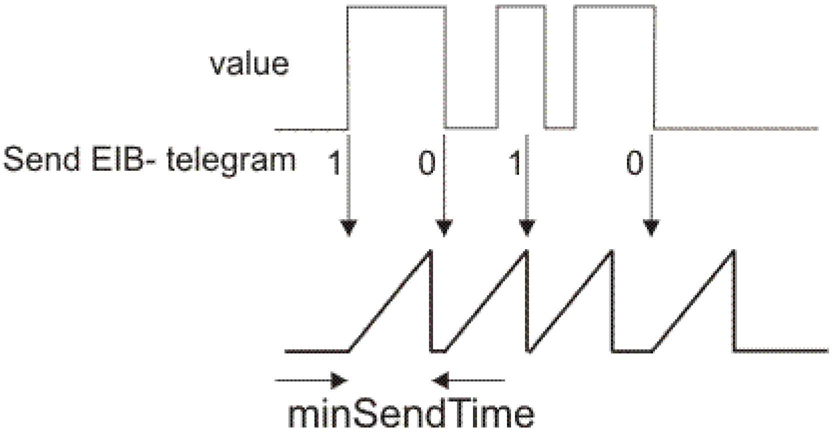
```
VAR_INPUT
  Group_Address : EIB_GROUP_ADDR;
  bControl      : BOOL;
  bValue        : BOOL;
  str_Rec       : EIB_REC;
END_VAR
```

Name	Type	Description
Group_Address	EIB_GROUP_ADDR [▶ 74]	Group address to which the data are sent. This group address must be entered in the filters!
bControl	BOOL	Value range TRUE/FALSE.
bValue	BOOL	Value range TRUE/FALSE.
str_Rec	EIB_REC [▶ 76]	The data structure with which the function block KL6301() [▶ 14] must be linked.

Outputs

```
VAR_OUTPUT
  bError    : BOOL;
  iErrorID  : EIB_ERROR_CODE;
END_VAR
```

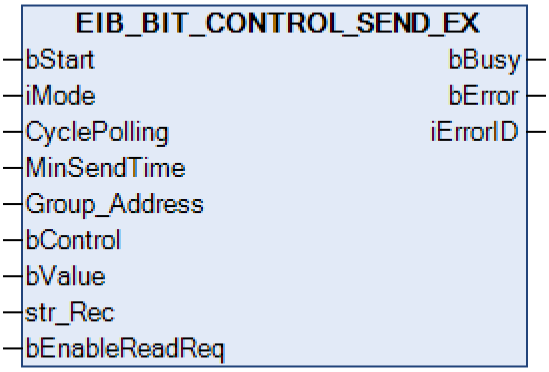
Name	Type	Description
bError	BOOL	This output goes TRUE as soon as an error occurs. This error is described via the *iErrorID* variable.
iErrorID	EIB_ERROR_CODE [▶ 72]	This output outputs an error code in the event of an error. *bError* goes TRUE at the same time.



Requirements

Development environment	required TC3 PLC library
TwinCAT from v3.1.4020.14	Tc2_EIB from 3.3.4.0

4.2.5.21 EIB_BIT_CONTROL_SEND_EX



This function block sends a 2-bit EIB value to the set group address. Two IEC61131-3 BOOL variables are available as input values. The data can be sent in Manual, Polling or OnChange depending on the set mode (*iMode*).

Inputs

```
VAR_INPUT
  bStart         : BOOL;
  iMode          : INT;
  CyclePolling   : TIME := t#500ms;
  MinSendTime    : TIME := t#1s;
  Group_Address  : EIB_GROUP_ADDR;
  bControl       : BOOL;
  bValue         : BOOL;
  str_Rec        : EIB_REC;
  bEnableReadReq : BOOL;
END_VAR
```

Name	Type	Description
bStart	BOOL	The function block is activated by a positive edge at this input.
iMode	INT	0 - With a positive edge at *bStart*, an EIB telegram is sent. If the output *bBusy* is FALSE again, the command is processed. 1 - Polling mode: If *bStart* is TRUE, EIB telegrams are sent at intervals of *CyclePolling*. 2 - OnChange mode: If *bStart* is TRUE, an EIB telegram is automatically sent when the data changes. *MinSendTime* can be used to parameterize the minimum interval between two EIB messages, in order to avoid excessive EIB network load. 3 - OnChangePolling mode: If *bStart* is TRUE, EIB telegrams are sent at intervals of *CyclePolling* or automatically when the data changes. The minimum interval between two EIB messages is set with *MinSendTime*.
CyclePolling	TIME	Polling time for *iMode* = 1 (polling mode). The minimum time is 200 ms.
MinSendTime	TIME	Minimum interval time, which has to elapse before a telegram is sent in OnChange mode. The minimum time is 200 ms.
Group_Address	EIB_GROUP_ADDR [▶ 74]	Group address to which the data are sent. This group address must be entered in the filters!
bControl	BOOL	Value range TRUE/FALSE.
bValue	BOOL	Value range TRUE/FALSE.
str_Rec	EIB_REC [▶ 76]	The data structure with which the function block KL6301() [▶ 14] must be linked.
bEnableReadReq	BOOL	Enables the execution of read commands.

Outputs

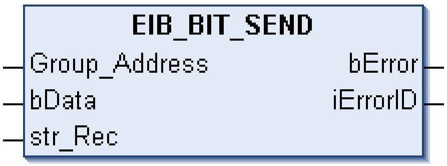
```
VAR_OUTPUT
  bBusy    : BOOL;
  bError   : BOOL;
  iErrorID : EIB_ERROR_CODE;
END_VAR
```

Name	Type	Description
bBusy	BOOL	The function block is active. Wait for new functions, until *bBusy* is FALSE again.
bError	BOOL	This output goes TRUE as soon as an error occurs. This error is described via the *iErrorID* variable.
iErrorID	EIB_ERROR_CODE [▶ 72]	This output outputs an error code in the event of an error. *bError* goes TRUE at the same time.

Requirements

Development environment	required TC3 PLC library
TwinCAT from v3.1.4020.14	Tc2_EIB from 3.3.4.0

4.2.5.22 EIB_BIT_SEND



This function block sends a 1-bit EIB value to the set group address. An IEC61131-3 BOOL variable is available as input value. The data are only transferred if there is a change in the data value. If the value changes again within 200 milliseconds, new data are only sent to the EIB device after another 200 millisecond has passed (see diagram). No new EIB telegram is sent if the value changes within the "min. send time" but falls back to the old, already sent value within the "min. send time".

Inputs

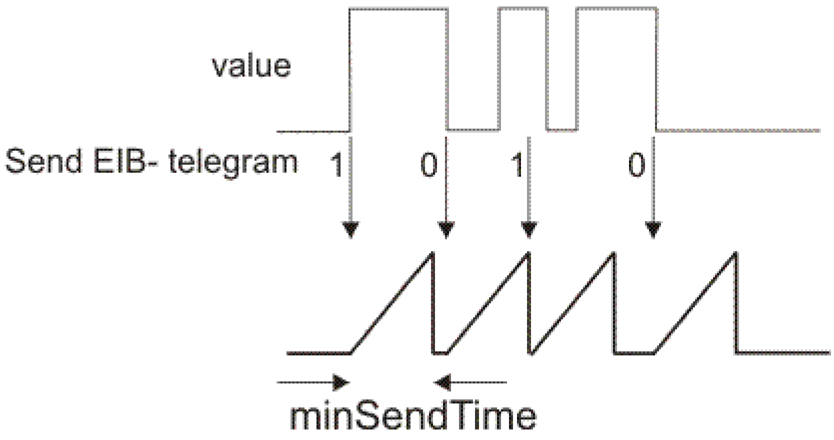
```
VAR_INPUT
  Group_Address  : EIB_GROUP_ADDR;
  bData          : BOOL;
  str_Rec        : EIB_REC;
END_VAR
```

Name	Type	Description
Group_Address	EIB_GROUP_ADDR [▶ 74]	Group address to which the data are sent. This group address must be entered in the filters!
bData	BOOL	Value range TRUE/FALSE.
str_Rec	EIB_REC [▶ 76]	The data structure with which the function block KL6301() [▶ 14] must be linked.

Outputs

```
VAR_OUTPUT
  bError   : BOOL;
  iErrorID : EIB_ERROR_CODE;
END_VAR
```

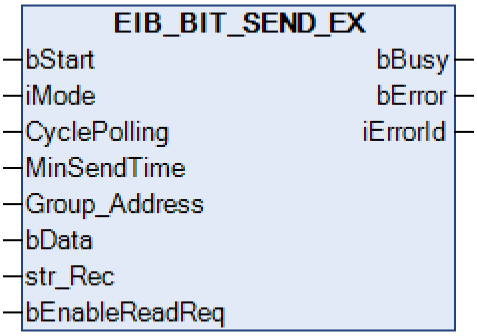
Name	Type	Description
bError	BOOL	This output goes TRUE as soon as an error occurs. This error is described via the *iErrorID* variable.
iErrorID	EIB_ERROR_CODE [▶ 72]	This output outputs an error code in the event of an error. *bError* goes TRUE at the same time.



Requirements

Development environment	required TC3 PLC library
TwinCAT from v3.1.4020.14	Tc2_EIB from 3.3.4.0

4.2.5.23 EIB_BIT_SEND_EX



This function block sends a Bool value to the set group address. The data can be sent in Manual, Polling or OnChange mode (*iMode*).

Inputs

```
VAR_INPUT
  bStart          : BOOL;
  iMode           : INT;
  CyclePolling    : TIME := t#10s;
  MinSendTime     : TIME := t#1s;
  Group_Address   : EIB_GROUP_ADDR;
  bData           : BOOL;
  str_Rec         : EIB_REC;
  bEnableReadReq  : BOOL;
END_VAR
```

Name	Type	Description
bStart	BOOL	Activates the function block. The function block starts operating depending on the set mode (see *iMode*).
iMode	INT	0 - With a positive edge at *bStart*, an EIB telegram is sent. If the output *bBusy* is FALSE again, the command is processed. 1 - Polling mode: If *bStart* is TRUE, EIB telegrams are sent at intervals of *CyclePolling*. 2 - OnChange mode: If *bStart* is TRUE, an EIB telegram is automatically sent when the data changes. *MinSendTime* can be used to parameterize the minimum interval between two EIB messages, in order to avoid excessive EIB network load. 3 - OnChangePolling mode: If *bStart* is TRUE, EIB telegrams are sent at intervals of *CyclePolling* or automatically when the data changes. The minimum interval between two EIB messages is set with *MinSendTime*.
CyclePolling	TIME	Polling time for *iMode* = 1 (polling mode). The minimum time is 200 ms.
MinSendTime	TIME	Minimum interval time, which has to elapse before a telegram is sent in OnChange mode. The minimum time is 200 ms.
Group_Address	EIB_GROUP_ADDR [▶ 74]	Group address from which the data were sent. This group address must be entered in the filters!
bData	BOOL	Value range TRUE/FALSE.
str_Rec	EIB_REC [▶ 76]	The data structure with which the function block KL6301() [▶ 14] must be linked.
bEnableReadReq	BOOL	Enables the execution of read commands.

Outputs

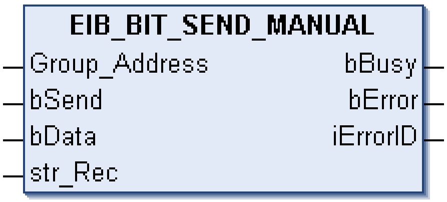
```
VAR_OUTPUT
  bBusy    : BOOL;
  bError   : BOOL;
  iErrorID : EIB_ERROR_CODE;
END_VAR
```

Name	Type	Description
bBusy	BOOL	The function block is active. Wait for new functions, until *bBusy* is FALSE again.
bError	BOOL	This output goes TRUE as soon as an error occurs. This error is described via the *iErrorID* variable.
iErrorID	EIB_ERROR_CODE [▶ 72]	This output outputs an error code in the event of an error. *bError* goes TRUE at the same time.

Requirements

Development environment	required TC3 PLC library
TwinCAT from v3.1.4020.14	Tc2_EIB from 3.3.4.0

4.2.5.24 EIB_BIT_SEND_MANUAL



This function block sends a 1-bit EIB value to the set group address. An IEC61131-3 BOOL variable is available as input value. The data is sent when *bSend* has a positive edge. *bBusy* is set as long the function block is active. *bBusy* is set to FALSE once the EIB command was sent or if an error occurs. An error is indicated by setting the *bError* variable. The error code is indicated in the *iErrorID*.

Inputs

```
VAR_INPUT
  Group_Address : EIB_GROUP_ADDR;
  bSend         : BOOL;
  bData         : BOOL;
  str_Rec       : EIB_REC;
END_VAR
```

Name	Type	Description
Group_Address	EIB_GROUP_ADDR [▶ 74]	Group address to which the data are sent. This group address must be entered in the filters!
bSend	BOOL	Positive edge sends EIB telegram..
bData	BOOL	Value range TRUE/FALSE.
str_Rec	EIB_REC [▶ 76]	The data structure with which the function block KL6301() [▶ 14] must be linked.

Outputs

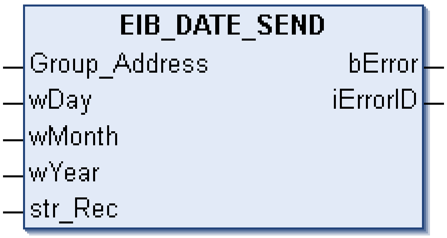
```
VAR_OUTPUT
  bBusy    : BOOL;
  bError   : BOOL;
  iErrorID : EIB_ERROR_CODE;
END_VAR
```

Name	Type	Description
bBusy	BOOL	The function block is active. Wait for new functions, until *bBusy* is FALSE again.
bError	BOOL	This output goes TRUE as soon as an error occurs. This error is described via the *iErrorID* variable.
iErrorID	EIB_ERROR_CODE [▶ 72]	This output outputs an error code in the event of an error. *bError* goes TRUE at the same time.

Requirements

Development environment	required TC3 PLC library
TwinCAT from v3.1.4020.14	Tc2_EIB from 3.3.4.0

4.2.5.25 EIB_DATE_SEND



This function block sends a 3-byte EIB value to the set group address. Three IEC61131-3 word variables are available as input values. The data are sent when the block is called for the first time the and then every 5 minutes.

Inputs

```
VAR_INPUT
  Group_Address : EIB_GROUP_ADDR;
  wDay          : WORD;
  wMonth          : WORD;
  wYear           : WORD;
  str_Rec         : EIB_REC;
END_VAR
```

Name	Type	Description
Group_Address	EIB_GROUP_ADDR [▸ 74]	Group address to which the data are sent. This group address must be entered in the filters!
wDay	WORD	Value range 1...31.
wMonth	WORD	Value range 1...12.
wYear	WORD	Value range 0...99. If a value greater 2000 entered, 2000 is automatically subtracted. For the year 2005, for example, only the 5 is transferred to the EIB node.
str_Rec	EIB_REC [▸ 76]	The data structure with which the function block KL6301() [▸ 14] must be linked.

Outputs

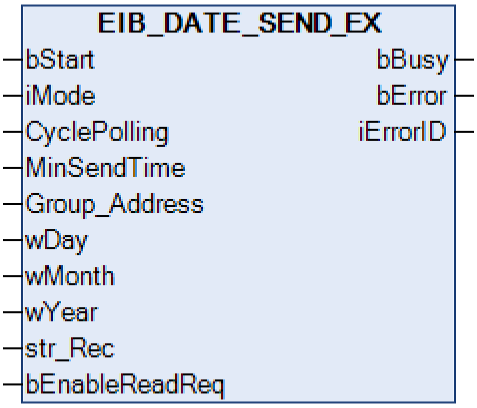
```
VAR_OUTPUT
  bError    : BOOL;
  iErrorID : EIB_ERROR_CODE;
END_VAR
```

Name	Type	Description
bError	BOOL	This output goes TRUE as soon as an error occurs. This error is described via the *iErrorID* variable.
iErrorID	EIB_ERROR_CODE [▸ 72]	This output outputs an error code in the event of an error. *bError* goes TRUE at the same time.

Requirements

Development environment	required TC3 PLC library
TwinCAT from v3.1.4020.14	Tc2_EIB from 3.3.4.0

4.2.5.26 EIB_DATE_SEND_EX



This function block sends a 3-byte EIB value to the set group address. Three IEC61131-3 word variables are available as input values. The data can be sent in Manual, Polling or OnChange depending on the set mode (*iMode*).

Inputs

```
VAR_INPUT
  bStart           : BOOL;
  iMode            : INT;
  CyclePolling     : TIME := t#500ms;
  MinSendTime      : TIME := t#1s;
  Group_Address    : EIB_GROUP_ADDR;
  wDay             : WORD;
  wMonth         : WORD;
  wYear          : WORD;
  str_Rec        : EIB_REC;
  bEnableReadReq : BOOL;
END_VAR
```

Name	Type	Description
bStart	BOOL	The function block is activated by a positive edge at this input.
iMode	INT	0 - With a positive edge at *bStart*, an EIB telegram is sent. If the output *bBusy* is FALSE again, the command is processed. 1 - Polling mode: If *bStart* is TRUE, EIB telegrams are sent at intervals of *CyclePolling*. 2 - OnChange mode: If *bStart* is TRUE, an EIB telegram is automatically sent when the data changes. *MinSendTime* can be used to parameterize the minimum interval between two EIB messages, in order to avoid excessive EIB network load. 3 - OnChangePolling mode: If *bStart* is TRUE, EIB telegrams are sent at intervals of *CyclePolling* or automatically when the data changes. The minimum interval between two EIB messages is set with *MinSendTime*.
CyclePolling	TIME	Polling time for *iMode* = 1 (polling mode). The minimum time is 200 ms.
MinSendTime	TIME	Minimum interval time, which has to elapse before a telegram is sent in OnChange mode. The minimum time is 200 ms.
Group_Address	EIB_GROUP_ADDR [▶ 74]	Group address to which the data are sent. This group address must be entered in the filters!
wDay	WORD	Value range 1...31.
wMonth	WORD	Value range 1...12.
wYear	WORD	Value range 0...99. If a value greater 2000 entered, 2000 is automatically subtracted. For the year 2005, for example, only the 5 is transferred to the EIB node.
str_Rec	EIB_REC [▶ 76]	The data structure with which the function block KL6301() [▶ 14] must be linked.
bEnableReadReq	BOOL	Enables the execution of read commands.

Outputs

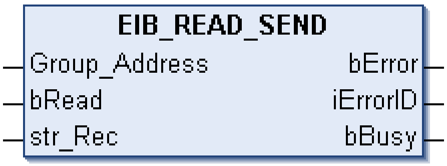
```
VAR_OUTPUT
  bBusy    : BOOL;
  bError   : BOOL;
  iErrorID : EIB_ERROR_CODE;
END_VAR
```

Name	Type	Description
bBusy	BOOL	The function block is active. Wait for new functions, until *bBusy* is FALSE again.
bError	BOOL	This output goes TRUE as soon as an error occurs. This error is described via the *iErrorID* variable.
iErrorID	EIB_ERROR_CODE [▶ 72]	This output outputs an error code in the event of an error. *bError* goes TRUE at the same time.

Requirements

Development environment	required TC3 PLC library
TwinCAT from v3.1.4024.57	Tc2_EIB from v3.4.3.0

4.2.5.27 EIB_READ_SEND



This function block sends a *Read_Group_Req* to the set group address. For receiving a Read_Group_Res the group address filter of the KL6301 has to be parameterized accordingly.

Inputs

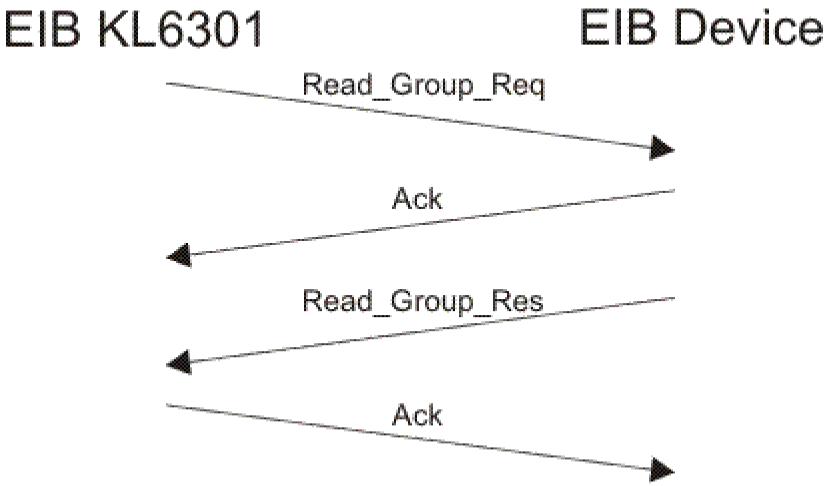
```
VAR_INPUT
  Group_Address : EIB_GROUP_ADDR;
  bRead         : BOOL;
  str_Rec       : EIB_REC;
END_VAR
```

Name	Type	Description
Group_Address	EIB_GROUP_ADDR [▶ 74]	Group address to which the data are sent. This group address must be entered in the filters!
bRead	BOOL	Positive edge starts the function block and sends a Read_Group_Req to the EIB device. To receive a response, the group address must be entered in the filter!
str_Rec	EIB_REC [▶ 76]	The data structure with which the function block KL6301() [▶ 14] must be linked.

Outputs

```
VAR_OUTPUT
  bError        : BOOL;
  iErrorID      : EIB_ERROR_CODE;
  bBusy         : BOOL;
END_VAR
```

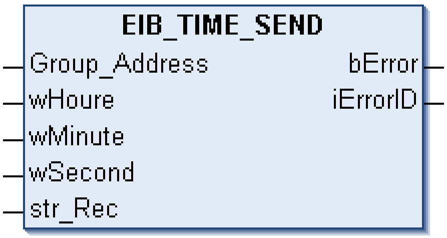
Name	Type	Description
bError	BOOL	This output goes TRUE as soon as an error occurs. This error is described via the *iErrorID* variable.
iErrorID	EIB_ERROR_CODE [▶ 72]	This output outputs an error code in the event of an error. *bError* goes TRUE at the same time.
bBusy	BOOL	The function block is in operation as long as *bBusy* is set, i.e. TRUE. Please wait until *bBusy* changes to FALSE.



Requirements

Development environment	required TC3 PLC library
TwinCAT from v3.1.4020.14	Tc2_EIB from 3.3.4.0

4.2.5.28 EIB_TIME_SEND



This function block sends a 3-byte EIB value to the set group address. Three IEC61131-3 word variables are available as input values. The data are sent when the block is called for the first time the and then every 5 minutes.

Inputs

```
VAR_INPUT
  Group_Address : EIB_GROUP_ADDR;
  wHoure        : WORD;
  wMinute       : WORD;
  wSecond       : WORD;
  str_Rec       : EIB_REC;
END_VAR
```

Name	Type	Description
Group_Address	EIB_GROUP_ADDR [▶ 74]	Group address to which the data are sent. This group address must be entered in the filters!
wHoure	WORD	Value range 0..23.
wMinute	WORD	Value range 0..59.
wSecond	WORD	Value range 0..59.
str_Rec	EIB_REC [▶ 76]	The data structure with which the function block KL6301() [▶ 14] must be linked.

Outputs

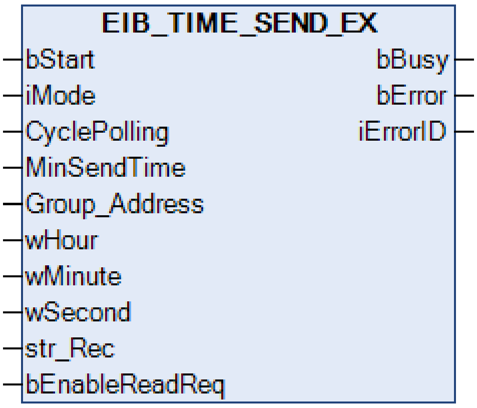
```
VAR_OUTPUT
  bError    : BOOL;
  iErrorID : EIB_ERROR_CODE;
END_VAR
```

Name	Type	Description
bError	BOOL	This output goes TRUE as soon as an error occurs. This error is described via the *iErrorID* variable.
iErrorID	EIB_ERROR_CODE [▶ 72]	This output outputs an error code in the event of an error. *bError* goes TRUE at the same time.

Requirements

Development environment	required TC3 PLC library
TwinCAT from v3.1.4020.14	Tc2_EIB from 3.3.4.0

4.2.5.29 EIB_TIME_SEND_EX



This function block sends a 3-byte EIB value to the set group address. Three IEC61131-3 word variables are available as input values. The data can be sent in Manual, Polling or OnChange depending on the set mode (*iMode*).

Inputs

```
VAR_INPUT
  bStart          : BOOL;
  iMode           : INT;
  CyclePolling    : TIME := t#500ms;
  MinSendTime     : TIME := t#1s;
  Group_Address   : EIB_GROUP_ADDR;
  wDay            : WORD;
  wMonth          : WORD;
  wYear           : WORD;
  str_Rec         : EIB_REC;
  bEnableReadReq : BOOL;
END_VAR
```

Name	Type	Description
bStart	BOOL	The function block is activated by a positive edge at this input.
iMode	INT	0 - With a positive edge at *bStart*, an EIB telegram is sent. If the output *bBusy* is FALSE again, the command is processed. 1 - Polling mode: If *bStart* is TRUE, EIB telegrams are sent at intervals of *CyclePolling*. 2 - OnChange mode: If *bStart* is TRUE, an EIB telegram is automatically sent when the data changes. *MinSendTime* can be used to parameterize the minimum interval between two EIB messages, in order to avoid excessive EIB network load. 3 - OnChangePolling mode: If *bStart* is TRUE, EIB telegrams are sent at intervals of *CyclePolling* or automatically when the data changes. The minimum interval between two EIB messages is set with *MinSendTime*.
CyclePolling	TIME	Polling time for *iMode* = 1 (polling mode). The minimum time is 200 ms.
MinSendTime	TIME	Minimum interval time, which has to elapse before a telegram is sent in OnChange mode. The minimum time is 200 ms.
Group_Address	EIB_GROUP_ADDR [▶ 74]	Group address to which the data are sent. This group address must be entered in the filters!
wHour	WORD	Value range 0..23.
wMinute	WORD	Value range 0..59.
wSecond	WORD	Value range 0..59.
str_Rec	EIB_REC [▶ 76]	The data structure with which the function block KL6301() [▶ 14] must be linked.
bEnableReadReq	BOOL	Enables the execution of read commands.

Outputs

```
VAR_OUTPUT
  bBusy    : BOOL;
  bError   : BOOL;
  iErrorID : EIB_ERROR_CODE;
END_VAR
```

Name	Type	Description
bBusy	BOOL	The function block is active. Wait for new functions, until *bBusy* is FALSE again.
bError	BOOL	This output goes TRUE as soon as an error occurs. This error is described via the *iErrorID* variable.
iErrorID	EIB_ERROR_CODE [▶ 72]	This output outputs an error code in the event of an error. *bError* goes TRUE at the same time.

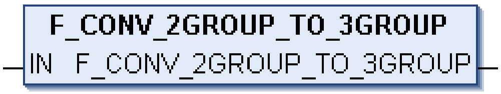
Requirements

Development environment	required TC3 PLC library
TwinCAT from v3.1.4024.57	Tc2_EIB from v3.4.3.0

4.2.6 Functions

Function blocks	Description
F_CONV_2GROUP_TO_3GROUP [▶ 69]	Conversion of a 2-stage group address to a 3-stage group address
F_CONV_3GROUP_TO_2GROUP [▶ 69]	Conversion of a 3-stage group address to a 2-stage group address

4.2.6.1 F_CONV_2GROUP_TO_3GROUP



Conversion of a 2-stage group address to a 3-stage group address.

Inputs

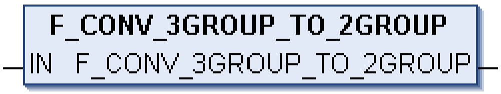
```
VAR_INPUT
  IN : EIB_GROUP_ADDR_2GROUP;
END_VAR
```

Name	Type	Description
Group_Address	EIB_GROUP_ADDR_2GROUP [▶ 75]	2-stage group address

Requirements

Development environment	required TC3 PLC library
TwinCAT from v3.1.4020.14	Tc2_EIB from 3.3.4.0

4.2.6.2 F_CONV_3GROUP_TO_2GROUP



Conversion of a 3-stage group address to a 2-stage group address.

Inputs

```
VAR_INPUT
  IN : EIB_GROUP_ADDR;
END_VAR
```

Name	Type	Description
IN	EIB_GROUP_ADDR [▶ 74]	3-stage group address

Requirements

Development environment	required TC3 PLC library
TwinCAT from v3.1.4020.14	Tc2_EIB from 3.3.4.0

4.2.7 Error codes

Value (hex)	Value (dec)	Value (enum)	Description
0x0000	0	NO_EIB_ERROR	No error.
0x0001	1	WRONG_EIB_PHYS_ADDR	Outdated, no longer used.
0x0002	2	WRONG_EIB_GROUP_ADDR	The input variable *EIB_GROUP_FILTER.GROUP_ADDR* is faulty. Check *GROUP_ADDR* of your filters. *MAIN* must be less than 16, *SUB_MAIN* less than 8.
0x0003	3	WRONG_EIB_GROUP_LEN	The input variable *EIB_GROUP_FILTER.GROUP_LEN* is faulty. Incorrect filter length. Check the mode and the length of the filters.
0x0004	4	WRONG_EIB_NO_FILTER	No filter detected. Check your filter in *EIB_GROUP_FILTER* and the mode.
0x0005	5	WRONG_EIB_IDX_RANGE	The input variable *idx* has an incorrect value.
0x000A	10	WRONG_EIB_FIRMWARE	The mode is not supported with this firmware.
0x000B	11	WRONG_EIB_MODE	Unsupported mode during parameterization. Check *iMode*. Permitted values are 0, 1 and 100.
0x000C	12	WRONG_MODE	The input variable *iMode* has an incorrect value.
0x000E	14	WRONG_EIB_FIRMWARE_B1_NECESSARY	Firmware B1 or higher required.
0x000F	15	WRONG_EIB_FIRMWARE_B3_NECESSARY	Firmware B3 or higher required.
0x0014	20	WRONG_EIB_DATA_LEN	Expected data length of the EIB telegram is wrong. Telegram is discarded. Check the EIB group addresses and/or the data type used.
0x0015	21	ERROR_EIB_SERVICE_NOT_SUPPORT	A EIB telegram is not supported.
0x001E	30	KL6301_TP_TOGGLE_ERROR	Terminal did not respond for one second. Check the connection with the KL6301. Is it still busy with data exchange?
0x001F	31	TIME_OUT	The terminal fails to respond during parameterization. Check the connection with the KL6301.
0x0020	32	KL6301_NO_RESPONSE_FROM_TERMINAL	No connection to KL6301. Either terminal not available or mapping incorrect.
0x0028	40	ERROR_SEND_8BIT_WRONG_Scaling_Mode	Incorrect or unsupported Scaling mode.
0x0064	100	ERROR_EIB_PHY_ADDR_NOT_SUPPORT	Physical addressing not permitted.
0x0065	101	ERROR_EIB_WRITE_DATA	Outdated. No longer used.
0x0066	102	MONITOR_MODE_LEN_IS_NOT_OK_MUST_0	For Monitor mode the filter length must be 0.
0x0067	103	MONITOR_MODE_ADDR_IS_NOT_OK_MUST_0	For Monitor mode the addresses must be 0.
0x0068	104	WATCHDOG_ERROR_NO_SEND	Data transfer not possible. The group address for which the data transfer has failed can be found in the local variable "NotSendGroup" of function block KL6301.
0x0BBB	3003	ERROR_EIB_NO_ACK	No ACK received.
0xFAFB	64251	ERROR_EIB_NO_COM_TO_TP	No communication with the EIB hardware.
0x0FCC	4044	ERROR_TP_TEMP_WARNING	Temperature in KL6301 exceeded.
0x17CC	6092	ERROR_TP_PROTOCOL_ERROR	Protocol error in EIB hardware.

Value (hex)	Value (dec)	Value (enum)	Description
0x27CC	10188	ERROR_TP_TRANSMITTER_ERROR	Protocol error in EIB hardware.
0x47CC	18380	ERROR_TP_RECEIVE_ERROR	Protocol error in EIB hardware.
0x87CC	34764	ERROR_TP_SLAVE_COLLISION	Too many collisions in the EIB hardware. Reduce the EIB load.

4.3 DUTs

4.3.1 Enums

Data types	Description
EIB_ERROR_CODE [▸ 72]	Error Messages
EIB_PRIORITY [▸ 74]	EIB telegram priority

4.3.1.1 EIB_ERROR_CODE

Library error messages.

```
TYPE EIB_ERROR_CODE :
(
  NO_EIB_ERROR                       := 0,
  WRONG_EIB_PHYS_ADDR                := 1,
  WRONG_EIB_GROUP_ADDR               := 2,
  WRONG_EIB_GROUP_LEN                := 3,
  WRONG_EIB_NO_FILTER                := 4,
  WRONG_EIB_IDX_RANGE                := 5,
  WRONG_EIB_FIRMWARE                 := 10,
  WRONG_EIB_MODE                     := 11,
  WRONG_MODE                         := 12,
  WRONG_EIB_FIRMWARE_B1_NECESSARY    := 14,
  WRONG_EIB_FIRMWARE_B3_NECESSARY    := 15,
  WRONG_EIB_DATA_LEN                 := 20,
  ERROR_EIB_SERVICE_NOT_SUPPORT      := 21,
  KL6301_TP_TOGGLE_ERROR             := 30,
  TIME_OUT                           := 31,
  KL6301_NO_RESPONSE_FROM_TERMINAL   := 32,
  ERROR_SEND_8BIT_WRONG_Scaling_Mode := 40,
  ERROR_EIB_PHY_ADDR_NOT_SUPPORT     := 100,
  ERROR_EIB_WRITE_DATA               := 101,
  MONITOR_MODE_LEN_IS_NOT_OK_MUST_0  := 102,
  MONITOR_MODE_ADDR_IS_NOT_OK_MUST_0 := 103,
  WATCHDOG_ERROR_NO_SEND             := 104,
  ERROR_EIB_NO_ACK                   := 16#0BBB,
  ERROR_EIB_NO_COM_TO_TP             := 16#FAFB,
  ERROR_TP_TEMP_WARNING              := 16#0FCC,
  ERROR_TP_PROTOCOL_ERROR            := 16#17CC,
  ERROR_TP_TRANSMITTER_ERROR         := 16#27CC,
  ERROR_TP_RECEIVE_ERROR             := 16#47CC,
  ERROR_TP_SLAVE_COLLISION           := 16#87CC
)
END_TYPE
```

Name	Description
NO_EIB_ERROR	No error.
WRONG_EIB_PHYS_ADDR.	Outdated, no longer used.
WRONG_EIB_GROUP_ADDR	The input variable *EIB_GROUP_FILTER.GROUP_ADDR* is faulty. Check *GROUP_ADDR* of your filters. *MAIN* must be less than 16, *SUB_MAIN* less than 8.
WRONG_EIB_GROUP_LEN	The input variable *EIB_GROUP_FILTER.GROUP_LEN* is faulty. Incorrect filter length. Check the mode and the length of the filters.
WRONG_EIB_NO_FILTER	No filter detected. Check your filter in *EIB_GROUP_FILTER* and the mode.
WRONG_EIB_IDX_RANGE	The input variable *idx* has an incorrect value.
WRONG_EIB_FIRMWARE	The mode is not supported with this firmware.
WRONG_EIB_MODE	Unsupported mode during parameterization. Check *iMode*. Permitted values are 0, 1 and 100.
WRONG_MODE	The input variable *iMode* has an incorrect value.
WRONG_EIB_FIRMWARE_B1_NECESSARY	Firmware B1 or higher required.
WRONG_EIB_FIRMWARE_B3_NECESSARY	Firmware B3 or higher required.
WRONG_EIB_DATA_LEN	Expected data length of the EIB telegram does not match. Telegram is discarded. Check the EIB group addresses and/or the data type used.
ERROR_EIB_SERVICE_NOT_SUPPORT	A EIB telegram is not supported.
KL6301_TP_TOGGLE_ERROR	Terminal did not respond for 1 second. Check the connection with the KL6301. Is it still busy with data exchange?
TIME_OUT	The terminal fails to respond during parameterization. Check the connection with the KL6301.
KL6301_NO_RESPONSE_FROM_TERMINAL	No connection to KL6301. Either the terminal does not exist or the mapping is wrong.
ERROR_SEND_8BIT_WRONG_Scaling_Mode	Incorrect or unsupported Scaling mode.
ERROR_EIB_PHY_ADDR_NOT_SUPPORT	Physical addressing not permitted.
ERROR_EIB_WRITE_DATA	Outdated, no longer used.
MONITOR_MODE_LEN_IS_NOT_OK_MUST_0	For Monitor mode the filter length must be 0.
MONITOR_MODE_ADDR_IS_NOT_OK_MUST_0	For Monitor mode the addresses must be 0.
WATCHDOG_ERROR_NO_SEND	Data transfer not possible. The group address for which the data transfer has failed can be found in the local variable "NotSendGroup" of function block KL6301.
ERROR_EIB_NO_ACK	No ACK received.
ERROR_EIB_NO_COM_TO_TP	No communication with the EIB hardware.
ERROR_TP_TEMP_WARNING	Temperature in KL6301 exceeded.
ERROR_TP_PROTOCOL_ERROR	Protocol error in EIB hardware.
ERROR_TP_TRANSMITTER_ERROR	Protocol error in EIB hardware.
ERROR_TP_RECEIVE_ERROR	Protocol error in EIB hardware.
ERROR_TP_SLAVE_COLLISION	Too many collisions in the EIB hardware. Reduce the EIB load.

Requirements

Development environment	required TC3 PLC library
TwinCAT from v3.1.4020.14	Tc2_EIB from 3.3.4.0

4.3.1.2 EIB_PRIORITY

EIB telegram priority

```
TYPE EIB_PRIORITY :
(
EIB_PRIORITY_LOW   := 1,
EIB_PRIORITY_HIGH  := 2,
EIB_PRIORITY_ALARM := 3
)
END_TYPE
```

Name	Description
EIB_PRIORITY_LOW	Priority low
EIB_PRIORITY_HIGH	Priority high
EIB_PRIORITY_ALARM	Priority alarm

Requirements

Development environment	required TC3 PLC library
TwinCAT from v3.1.4020.14	Tc2_EIB from 3.3.4.0

4.3.2 Structure

Data types	Description
EIB_GROUP_ADDR [▸ 74]	3-stage group address
EIB_GROUP_ADDR_2GROUP [▸ 75]	2-stage group address
EIB_GROUP_FILTER [▸ 75]	Group filter
EIB_PHYS_ADDR [▸ 75]	Physical address
EIB_REC [▸ 76]	Links the send and receive blocks with the block *KL6301*

4.3.2.1 EIB_GROUP_ADDR

3-stage group address

```
TYPE EIB_GROUP_ADDR :
STRUCT
  MAIN     : BYTE;
  SUB_MAIN : BYTE;
  NUMBER   : BYTE;
END_STRUCT
END_TYPE
```

Name	Description
MAIN	Main group (value range 0..31)
SUB_MAIN	Middle group (value range 0..7)
NUMBER	Subgroup (value range 0..255)

Requirements

Development environment	required TC3 PLC library
TwinCAT from v3.1.4020.14	Tc2_EIB from 3.3.4.0

4.3.2.2 EIB_GROUP_ADDR_2GROUP

2-stage group address

```
TYPE EIB_GROUP_ADDR_2GROUP :
STRUCT
  MAIN     : BYTE;
  SUB_MAIN : WORD;
END_STRUCT
END_TYPE
```

Name	Description
MAIN	Main group (value range 0..15)
SUB_MAIN	Subgroup (value range 0..2048)

Requirements

Development environment	required TC3 PLC library
TwinCAT from v3.1.4020.14	Tc2_EIB from 3.3.4.0

4.3.2.3 EIB_GROUP_FILTER

Group filter

```
TYPE EIB_GROUP_FILTER :
STRUCT
  GROUP_ADDR : EIB_GROUP_ADDR;
  GROUP_LEN  : WORD;
END_STRUCT
END_TYPE
```

Name	Description
GROUP_ADDR	Group address (see EIB_GROUP_ADDR [▶ 74])
GROUP_LEN	iMode 0 - 0..63, iMode 1 - 0..31

Requirements

Development environment	required TC3 PLC library
TwinCAT from v3.1.4020.14	Tc2_EIB from 3.3.4.0

4.3.2.4 EIB_PHYS_ADDR

Physical address

```
TYPE EIB_PHYS_ADDR :
STRUCT
  Area   : BYTE := 1;
  Line   : BYTE := 2;
  Device : BYTE := 3;
END_STRUCT
END_TYPE
```

Name	Description
Area	0..15
Line	0..15
Device	0..255

Requirements

Development environment	required TC3 PLC library
TwinCAT from v3.1.4020.14	Tc2_EIB from 3.3.4.0

4.3.2.5 EIB_REC

Links the send and receive function blocks with the function block *KL6301.*

```
TYPE EIB_REC :
STRUCT
  Rec_Group       : EIB_GROUP_ADDR;
  Rec_Len         : INT;
  Rec_Idx         : INT := 1;
  Rec_Data        : ARRAY[1..15] OF BYTE;
  Rec_bWriteBusy : BOOL;
  Rec_bReadBusy  : BOOL;
  Rec_bReady      : BOOL;
  Rec_bError      : BOOL;
  Rec_iErrorID    : EIB_Error_Code;
  pStr_Send       : PVOID;
  Rec_Data_rec    : BOOL;
  Rec_Typ         : EIB_Read_Typ;
END_STRUCT
END_TYPE
```

Name	Description
Rec_Group	Group address (see EIB_GROUP_ADDR)
Rec_Len	Length
Rec_Idx	Index
Rec_Data	Data bytes
Rec_bWriteBusy	Data is written
Rec_bReadBusy	Data is read
Rec_bReady	Ready
Rec_bError	This output goes TRUE as soon as an error occurs. This error is described via the *Rec_iErrorID* variable.
Rec_iErrorID	This output outputs an error code in the event of an error (see EIB_ERROR_CODE). *bError* goes TRUE at the same time.
pStr_Send	Pointer to the data to be sent.
Rec_Data_rec	Signals reception of data.
Rec_Typ	Type of telegram.

Requirements

Development environment	required TC3 PLC library
TwinCAT from v3.1.4020.14	Tc2_EIB from 3.3.4.0

4.4 Integration into TwinCAT

4.4.1 KL6301 with CX5120

This example describes how a simple PLC program for EIB can be written in TwinCAT and how it is linked with the hardware. The task is to change the state of a switching output with a button.

Example: https://infosys.beckhoff.com/content/1033/tcplclib_tc2_eib/Resources/6165821835/.zip

Hardware

Setting up the components

- 1x CX5120 Embedded PC
- 1x KL1104 four-channel digital input terminal (for the Set/Reset function)
- 1x KL6301 EIB terminal
- 1x KL9010 end terminal

Set up the hardware and the EIB components as described in the associated documentation.

This example assumes that a Set button was connected to the first KL1104 input and a Reset button to the second, and that the EIB group address of the switching output is known.

Software

Creation of the PLC program

Create a new "TwinCAT XAE Project" and create a "Standard PLC Project".
Add the Tc2_EIB library to the PLC project at "References".
Create a global variable list with the name GVL_EIB and create the following variables:

```
VAR_GLOBAL
    bSet                AT %I* : BOOL;
    bReset              AT %I* : BOOL
    arrKL6301_IN        AT %I* : ARRAY[1..24] OF BYTE;
    arrKL6301_OUT       AT %Q* : ARRAY[1..24] OF BYTE;
    stDataRec                  : EIB_REC;
END_VAR
```

Name	Type	Description
bSet	BOOL	Input variable for the Set button.
bReset	BOOL	Input variable for the Reset button.
arrKL6301_IN	BYTE	Input variable for the EIB terminal.
arrKL6301_OUT	BYTE	Output variable for the EIB terminal.
stDataRec	EIB_REC [▶ 76]	Structure for the communication with EIB.

All EIB function blocks must be called in the same task.

In the MAIN program (ST), call the function block KL6301 [▶ 14] and create the following variables.

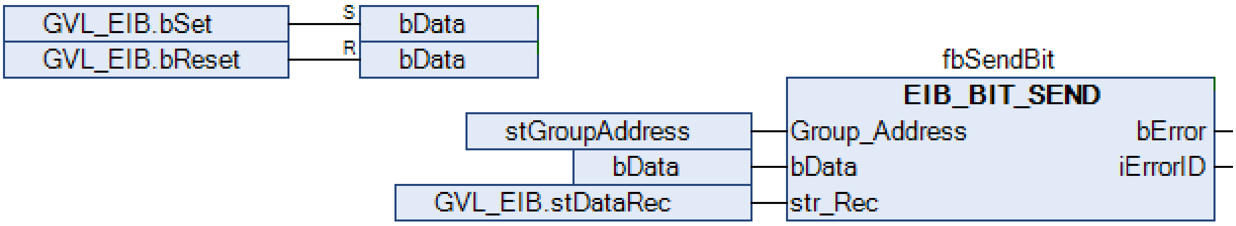
```
PROGRAM MAIN
VAR
    fbKL6301       : KL6301;
    iStep          : INT;
END_VAR
```

In the program part, the KL6301 is initially assigned a physical address and the filter for receiving EIB telegrams is configured. For more information see KL6301 [▶ 14].

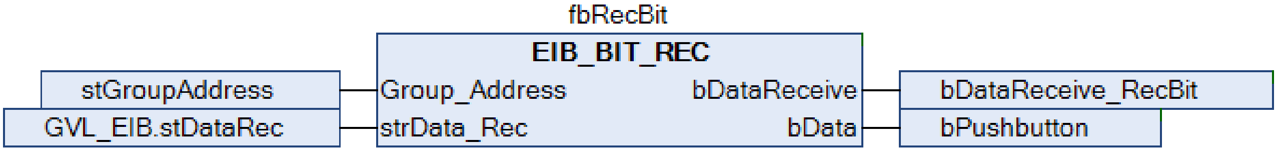
```
CASE iStep OF
0:  // Initialising of the physical EIB address; has to be unique in the network
    fbKL6301.EIB_PHYS_ADDR.Area := 2; //EIB Address 2.3.4
    fbKL6301.EIB_PHYS_ADDR.Line := 3;
    fbKL6301.EIB_PHYS_ADDR.Device := 4;
// Configuration of filters for the group addresses
// Filter 1 1/1/0 LEN 63
    fbKL6301.EIB_GROUP_FILTER[1].GROUP_ADDR.MAIN:=1;
    fbKL6301.EIB_GROUP_FILTER[1].GROUP_ADDR.SUB_MAIN:=1;
    fbKL6301.EIB_GROUP_FILTER[1].GROUP_ADDR.NUMBER:=0;
    fbKL6301.EIB_GROUP_FILTER[1].GROUP_LEN:=63;
// Filter 2 1/2/0 LEN 63
    fbKL6301.EIB_GROUP_FILTER[2].GROUP_ADDR.MAIN:=1;
    fbKL6301.EIB_GROUP_FILTER[2].GROUP_ADDR.SUB_MAIN:=2;
    fbKL6301.EIB_GROUP_FILTER[2].GROUP_ADDR.NUMBER:=0;
    fbKL6301.EIB_GROUP_FILTER[2].GROUP_LEN:=63;
//Filter 3 1/4/0 LEN 63
    fbKL6301.EIB_GROUP_FILTER[3].GROUP_ADDR.MAIN:=1;
    fbKL6301.EIB_GROUP_FILTER[3].GROUP_ADDR.SUB_MAIN:=4;
    fbKL6301.EIB_GROUP_FILTER[3].GROUP_ADDR.NUMBER:=0;
    fbKL6301.EIB_GROUP_FILTER[3].GROUP_LEN:=63;
    //Filter 4 3/1/0 LEN 63
    fbKL6301.EIB_GROUP_FILTER[4].GROUP_ADDR.MAIN:=3;
    fbKL6301.EIB_GROUP_FILTER[4].GROUP_ADDR.SUB_MAIN:=1;
    fbKL6301.EIB_GROUP_FILTER[4].GROUP_ADDR.NUMBER:=0;
    fbKL6301.EIB_GROUP_FILTER[4].GROUP_LEN:=63;
// bActivate: activates the configuration of the KL6301 and enables the further data exchange
    fbKL6301.bActivate :=TRUE;

    IF fbKL6301.bReady THEN
        IF NOT fbKL6301.bError THEN
            iStep:=1; // EIB functionblock is now ready to send and receive
        ELSE
            iStep:=100; // EIB ERROR
        END_IF
    END_IF
1:  ;      // EIB functionblock is now ready to send and receive
100:  ;      // optional error handling EIB ERROR
END_CASE
//Call communication function block
fbKL6301(
    iMode:= 0, // 4 filter with 64 entries each
    KL6301_IN:= GVL_EIB.arrKL6301_in,
    KL6301_OUT:= GVL_EIB.arrKL6301_out,
    str_Data_Rec => GVL_EIB.stDataRec);

//Call programs for reading and sending
EIB_Send_Prg();
EIB_Rec_Prg();
```

Create an EIB program called EIB_Send_Prg() (CFC), in which the function block EIB_BIT_SEND [▶ 59] is called.

Link the local variable *bData* with the global variables *bSet* and *bReset*, then with the input *bData* of the send block. Link the global variable *stDataRec* with *st_Rec*.

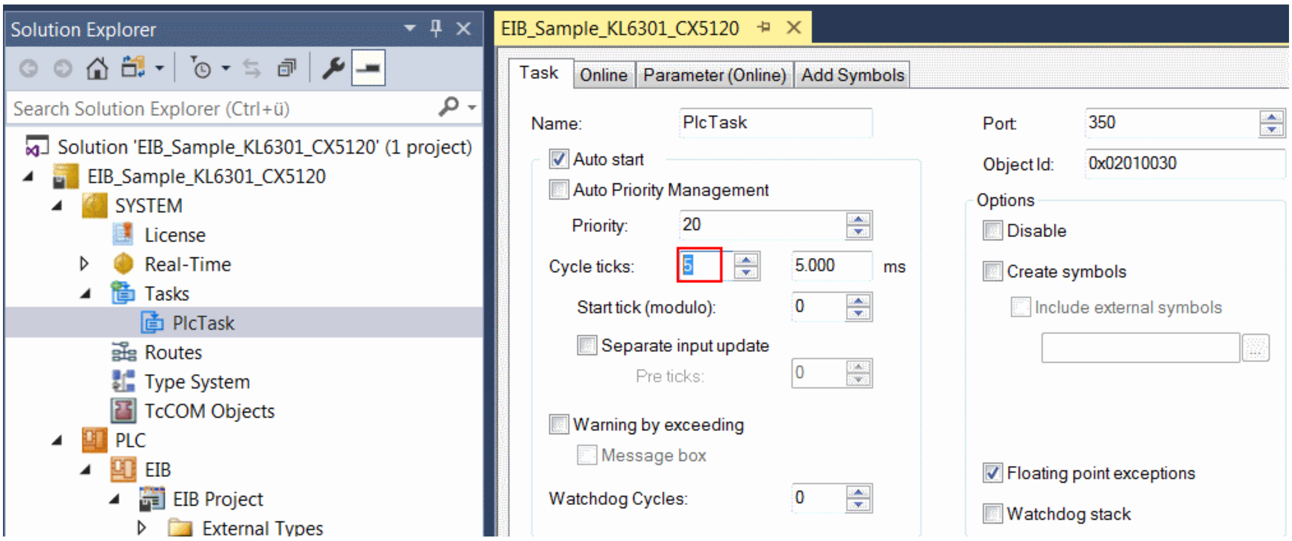


Create an EIB program called EIB_Rec_Prg() (CFC), in which the function block EIB_BIT_REC [▶ 27] is called. In the sample, the pressing of a button is read by the function block. Link the global variable *stDataRec* with *st_Rec*.



Go to the task configuration and give the task a lower cycle time of 5 ms, for example.

Further conditions can be found in the description of the function block KL6301 [▶ 14].



Name: PlcTask

Port: 350

Object Id: 0x02010030

☑ Auto start

☐ Auto Priority Management

Priority: 20

Cycle ticks: 5 5.000 ms

Start tick (modulo): 0

☐ Separate input update

Pre ticks: 0

☐ Warning by exceeding

☐ Message box

Watchdog Cycles: 0

Options

☐ Disable

☐ Create symbols

☐ Include external symbols

☑ Floating point exceptions

☐ Watchdog stack

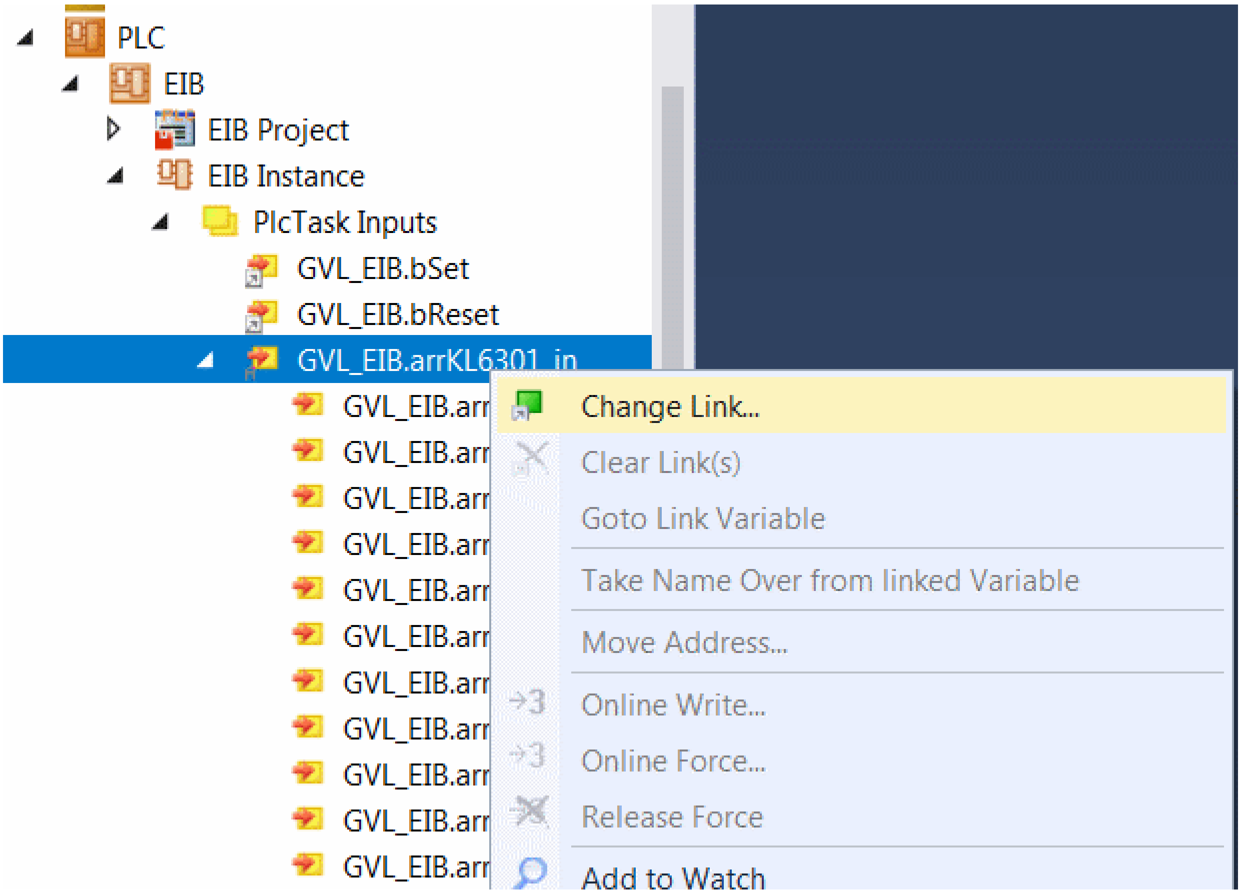
I/O configuration

Select the CX as target system and initiate a search for its hardware. In the project instance within the PLC section, you can see that the input and output variables are assigned to the task.

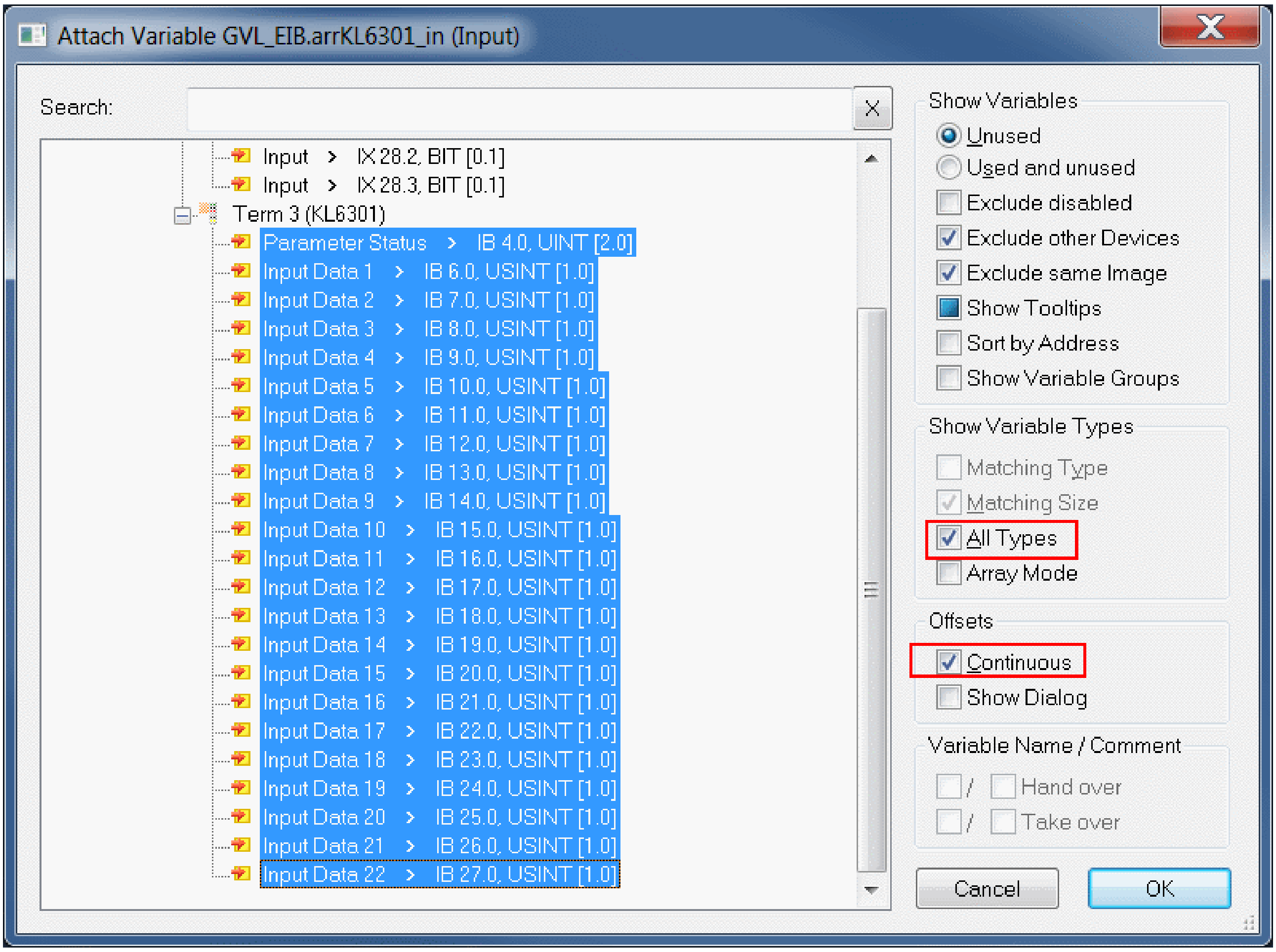
Now link the global variables with the inputs and outputs of the bus terminals.

The linking of the EIB variables is described in detail below.

Right-click the array *arrKL6301_in* and select "Change Link".



Under “I/O Configuration” select the terminal, select “All Types” and “Continuous”, then select “ParameterStatus”, “InputData1” to “InputData22” with the left mouse button and the >SHIFT< key. Then click "OK".



Attach Variable GVL_EIB.arrKL6301_in (Input)

Search:

```
Input > IX 28.2, BIT [0.1]
Input > IX 28.3, BIT [0.1]
Term 3 (KL6301)
    Parameter Status > IB 4.0, UINT [2.0]
    Input Data 1 > IB 6.0, USINT [1.0]
    Input Data 2 > IB 7.0, USINT [1.0]
    Input Data 3 > IB 8.0, USINT [1.0]
    Input Data 4 > IB 9.0, USINT [1.0]
    Input Data 5 > IB 10.0, USINT [1.0]
    Input Data 6 > IB 11.0, USINT [1.0]
    Input Data 7 > IB 12.0, USINT [1.0]
    Input Data 8 > IB 13.0, USINT [1.0]
    Input Data 9 > IB 14.0, USINT [1.0]
    Input Data 10 > IB 15.0, USINT [1.0]
    Input Data 11 > IB 16.0, USINT [1.0]
    Input Data 12 > IB 17.0, USINT [1.0]
    Input Data 13 > IB 18.0, USINT [1.0]
    Input Data 14 > IB 19.0, USINT [1.0]
    Input Data 15 > IB 20.0, USINT [1.0]
    Input Data 16 > IB 21.0, USINT [1.0]
    Input Data 17 > IB 22.0, USINT [1.0]
    Input Data 18 > IB 23.0, USINT [1.0]
    Input Data 19 > IB 24.0, USINT [1.0]
    Input Data 20 > IB 25.0, USINT [1.0]
    Input Data 21 > IB 26.0, USINT [1.0]
    Input Data 22 > IB 27.0, USINT [1.0]
```

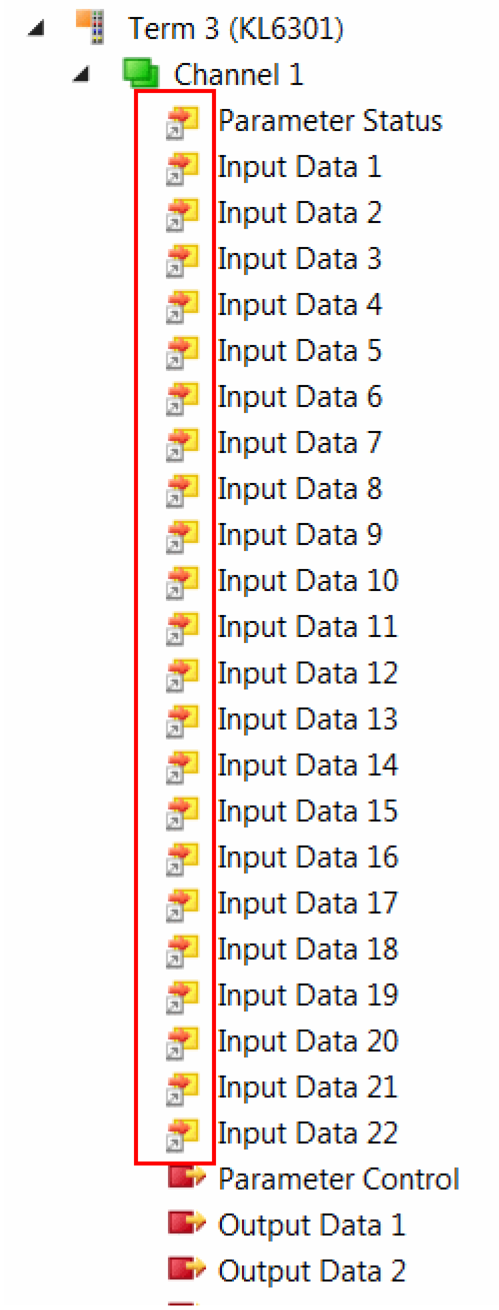
Show Variables
- ● Unused
- ○ Used and unused
- □ Exclude disabled
- ☑ Exclude other Devices
- ☑ Exclude same Image
- ■ Show Tooltips
- □ Sort by Address
- □ Show Variable Groups

Show Variable Types
- □ Matching Type
- ☑ Matching Size
- ☑ All Types
- □ Array Mode

Offsets
- ☑ Continuous
- □ Show Dialog

Variable Name / Comment
- □ / □ Hand over
- □ / □ Take over

You can now check the link. To do this go onto the KL6301 and open it. All terminal data should now show a small arrow. If that is the case, then proceed in exactly the same way with the outputs.



Term 3 (KL6301)

Channel 1

Parameter Status

Input Data 1

Input Data 2

Input Data 3

Input Data 4

Input Data 5

Input Data 6

Input Data 7

Input Data 8

Input Data 9

Input Data 10

Input Data 11

Input Data 12

Input Data 13

Input Data 14

Input Data 15

Input Data 16

Input Data 17

Input Data 18

Input Data 19

Input Data 20

Input Data 21

Input Data 22

Parameter Control

Output Data 1

Output Data 2

5 Appendix

5.1 Support and Service

Beckhoff and their partners around the world offer comprehensive support and service, making available fast and competent assistance with all questions related to Beckhoff products and system solutions.

Download finder

Our download finder contains all the files that we offer you for downloading. You will find application reports, technical documentation, technical drawings, configuration files and much more.

The downloads are available in various formats.

Beckhoff's branch offices and representatives

Please contact your Beckhoff branch office or representative for local support and service on Beckhoff products!

The addresses of Beckhoff's branch offices and representatives round the world can be found on our internet page: www.beckhoff.com

You will also find further documentation for Beckhoff components there.

Beckhoff Support

Support offers you comprehensive technical assistance, helping you not only with the application of individual Beckhoff products, but also with other, wide-ranging services:

- support
- design, programming and commissioning of complex automation systems
- and extensive training program for Beckhoff system components

Hotline:	+49 5246 963-157
e-mail:	support@beckhoff.com

Beckhoff Service

The Beckhoff Service Center supports you in all matters of after-sales service:

- on-site service
- repair service
- spare parts service
- hotline service

Hotline:	+49 5246 963-460
e-mail:	service@beckhoff.com

Beckhoff Headquarters

Beckhoff Automation GmbH & Co. KG

Huelshorstweg 20
33415 Verl
Germany

Phone:	+49 5246 963-0
e-mail:	info@beckhoff.com
web:	www.beckhoff.com

More Information:
www.beckhoff.com/te1000

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