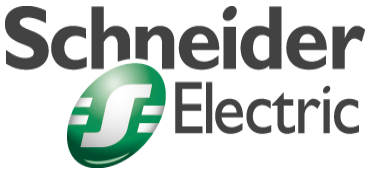
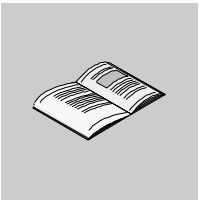
Concept
IEC block library
Part: ANA_IO
Volume 1

840 USE 504 00 eng Version 2.6



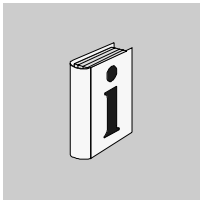
33002209.00

Table of Contents



The chapters marked gray are not included in this volume.

About the book



At a Glance

Document Scope This documentation is designed to help with the configuration of functions and function blocks.

Validity Note This documentation applies to Concept 2.6 under Microsoft Windows 98, Microsoft Windows 2000, Windows XP and Microsoft Windows NT 4.x.

Note: There is additional up to date tips in the README data file in Concept.

Related Documents

Title of Documentation	Reference Number
Concept Installation Instructions	840 USE 502 00
Concept User Manual	840 USE 503 00
Concept EFB User Manual	840 USE 505 00
Concept LL984 Block Library	840 USE 506 00

User Comments We welcome your comments about this document. You can reach us by e-mail at TECHCOMM@modicon.com

General information on the ANA_IO block library

I

Overview

At a Glance

This section contains general information on the ANA_IO block library.

What's in this part?

This part contains the following chapters:

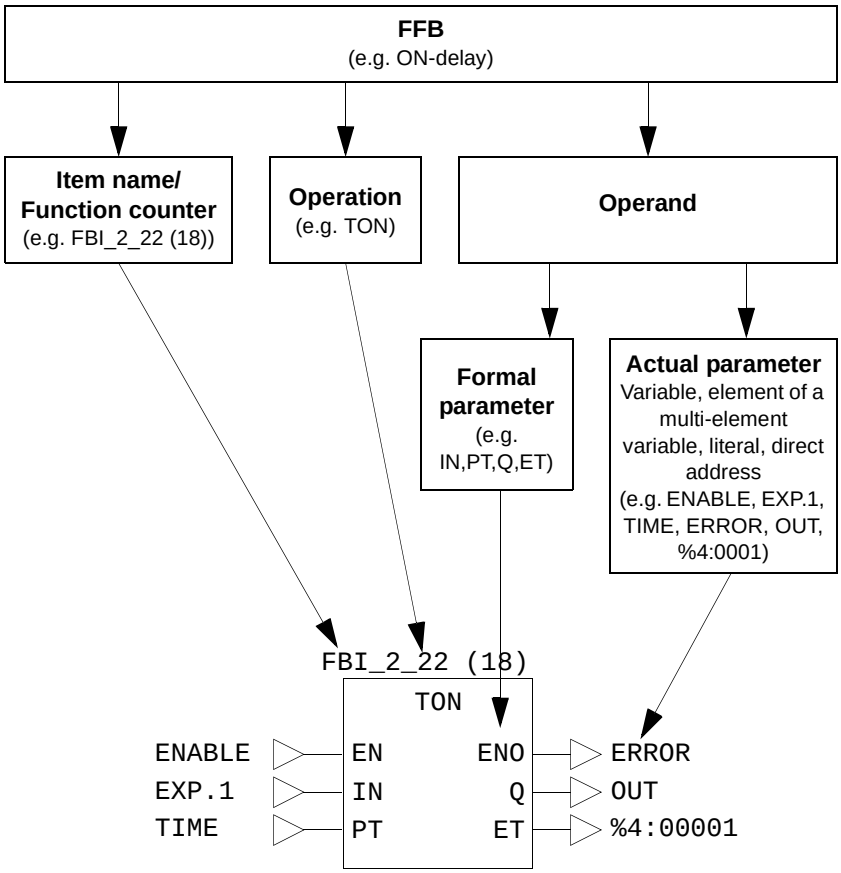
Parameterizing functions and function blocks

1

Parameterizing functions and function blocks

General

Each FFB consists of an operation, the operands needed for the operation and an instance name or function counter.



Operation

The operation determines which function is to be executed with the FFB, e.g. shift register, conversion operations.

Operand

The operand specifies what the operation is to be executed with. With FFBs, this consists of formal and actual parameters.

Formal/actual parameters

The formal parameter holds the place for an operand. During parameterization, an actual parameter is assigned to the formal parameter.

The actual parameter can be a variable, a multi-element variable, an element of a multi-element variable, a literal or a direct address.

Conditional/ unconditional calls

"Unconditional" or "conditional" calls are possible with each FFB. The condition is realized by pre-linking the input EN.

- Displayed EN
  conditional calls (the FFB is only processed if EN = 1)
- EN not displayed
  unconditional calls (FFB is always processed)

Note: If the EN input is not parameterized, it must be disabled. Any input pin that is not parameterized is automatically assigned a "0" value. Therefore, the FFB should never be processed.

Calling functions and function blocks in IL and ST

Information on calling functions and function blocks in IL (Instruction List) and ST (Structured Text) can be found in the relevant chapters of the user manual.

Operating Analog Modules

2

Overview

At a Glance

This block library contains EFBs to operate the analog modules and EFBs to operate the INTERBUS on the Compact.

Note: This block library's EFBs are available in every IEC application. It is not possible to use these platform specific EFBs on a PLC platform for which they were not intended (Quantum EFBs on a Compact PLC).

The EFBs for the analog modules can be found in the following groups:

- Quantum I/O Config
- Compact I/O Config
- Analog I/O Config
- Analog I/O Scaling
- Analog I/O Debug

What's in this chapter?

This chapter contains the following topics:

Editing analog values

Introduction

This block library contains EFBs for the operation of analog modules. The EFBs are designed in such a way that it enables the FBD program configuration to be largely independent of the hardware module used. Hardware-dependent EFBs (e.g. Group: Quantum IO Config) are used to evaluate project-specific information on the PLC and to process it in the data structures ANL_IN und ANL_OUT. Hardware-independent EFBs operate with these data structures to read the raw values from the Input words (3x), scale them and convert them into REAL values. This means that changes in direct addresses or changes in input/output parameters can be detected automatically by the EFBs.

Division into sections

As the detection of configuration data only occurs once after loading, it is advisable to divide the EFBs in the ANA_I/O library into at least two sections.

Division into at least two sections is recommended.
- Scaling section
- Configuration section

This division into a configuration section and several scaling sections can lead to a reduction of the CPU load, as the configuration part (configuration section) must only execute once, (after a cold restart or a warm restart). As a rule, the scaling sections must be executed continuously.

Scaling and configuration sections

Scaling section

Scaling sections are used for actual analog value processing.

Configuration section

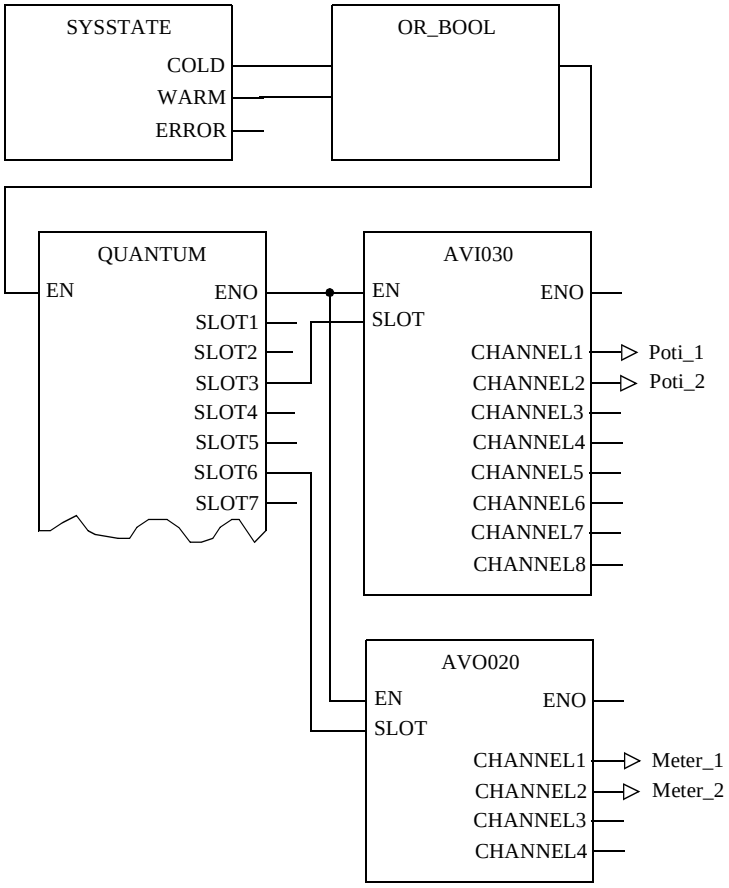
The configuration section is used to configure the analog I/O modules and controls the data exchange between analog EFBs, the state RAMand the configuration data. The configuration section should be called "CfgAnaIo" to ensure compatibility with future Concept versions.

Two options are available to control a configuration section:
- using the EN inputs of the individual EFBs
- by enabling or disabling the configuration section

Example 1 Control using the EN inputs

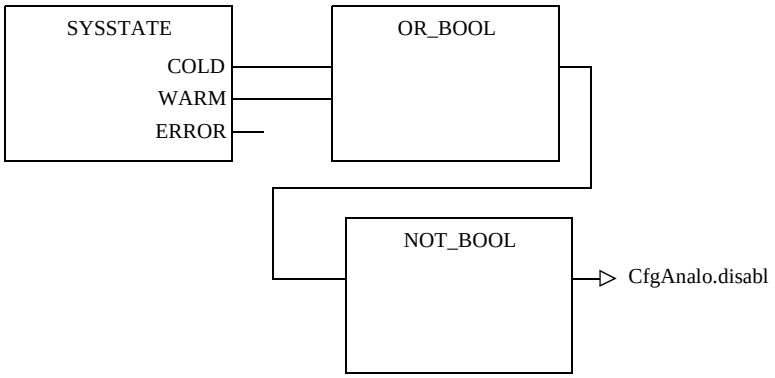
Control of the configuration section is possible through the EN inputs of this section's individual EFBs. The EFBs are enabled through the SYSSTATE EFB which has COLD or WARM outputs that are set to 1 for one cycle after either a cold or a warmstart.

Example of a configuration section "CfgAnaIo" for Quantum

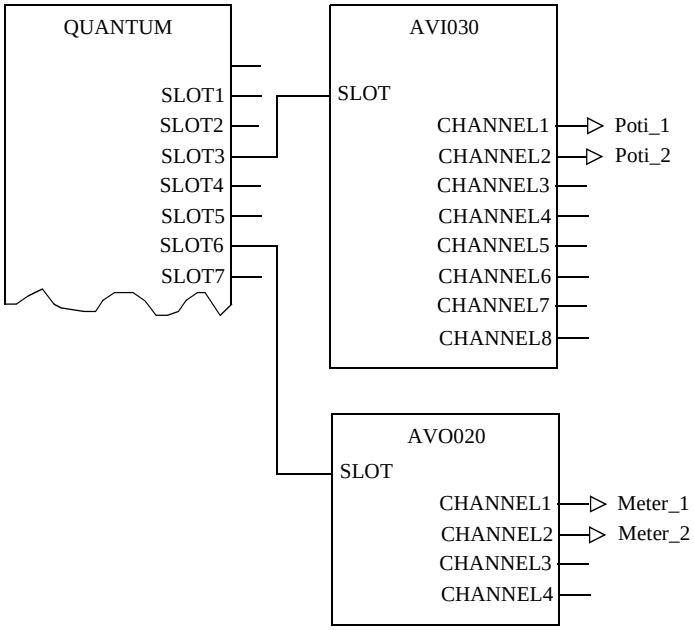


Example 2 Control by section enabling

The configuration section can be controlled by enabling and disabling this section. The configuration section is enabled in a section of its own through the SYSSTATE EFB which has COLD or WARM outputs that are set to 1 for one cycle after either a cold or a warmstart. This 1-signal is used for configuration section enable and disable. An EN and ENO linking of the EFBs is not required in this solution.
Example of a control section "Config_Ctrl" for Quantum



Example of a configuration section "CfgAnaIo" for Quantum



Configuration EFBs

At a Glance

Configuration data from analog input/output modules is accessed using EFBs from the block library groups "Quantum IO Config" and "Compact IO Config".
A difference is made in the following cases:

- *Procedure for only local I/O (Quantum/Compact), p. 11*
- *Procedure for expansion of the local backplane using XBE modules (Quantum), p. 13*
- *Procedure for distributed I/O (RIO, DIO) (Quantum), p. 14*
- *Procedure for expansion of the distributed backplane using XBE modules (Quantum), p. 15*

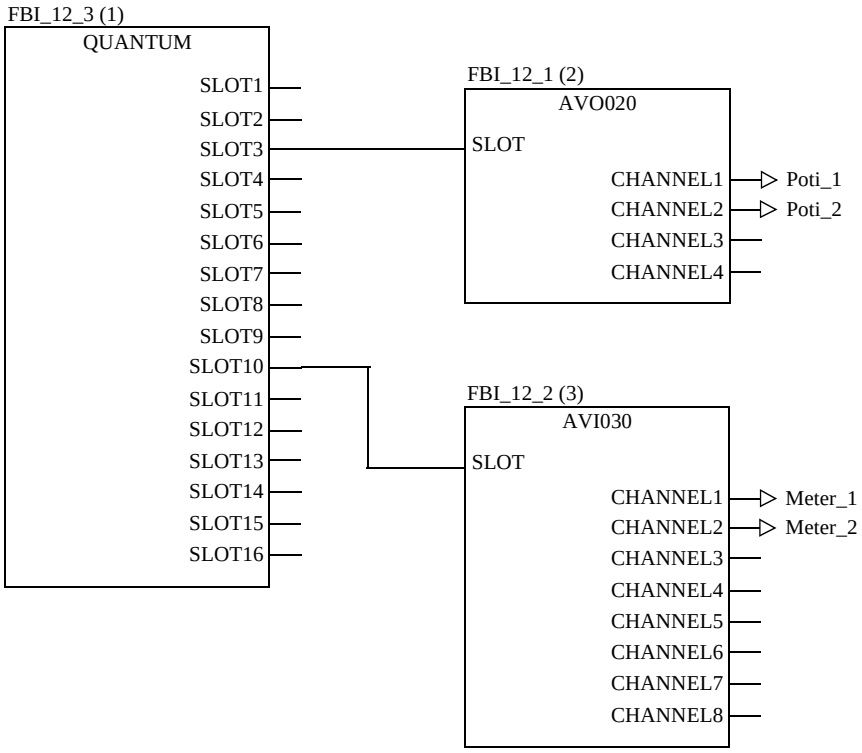
Procedure for only local I/O (Quantum/ Compact)

Place a single QUANTUM/COMPACT-EFB in the configuration section (CfgAnaIo). Each analog module has its own analog I/O EFB. Connect the desired analog I/O EFB to the corresponding slot on the QUANTUM/COMPACT EFB. The analog I/O EFB provides a variable of data type ANL_IN or ANL_OUTas an output. These values are further processed using the scaling EFBs in the scaling sections. To do this, they are connected with the relevant scaling EFBs using Unlocated variables .

Note: Do not specify Literals at the SLOT inputs of the configuration EFBs. SLOT inputs must be connected to SLOT outputs.

Example of only local I/O

Example of a configuration section "CfgAnaIo"



The EFB's mode of functioning can be found in the table below.

EFB	Function mode
Quantum	The EFB is used to edit the configuration data of a primary subrack for transfer by the analog in/out EFBs.
AVI030	Quantum module AVI 030 00 configuration. The EFB is used to edit the configuration data of the Quantum module AVI 030 00 for continued processing by the scaling EFBs. The 3x references specified in the I/O map are automatically assigned internally to the individual channels and can therefore only be occupied by Unlocated variables.
AVO020	Quantum module AVO 020 00 configuration. The EFB is used to edit the configuration data of the Quantum module AVO 020 00 for continued processing by the scaling EFBs. The 3x references and 4x references specified in the I/O map are automatically assigned internally to the individual channels and can therefore only be occupied by Unlocated variables.

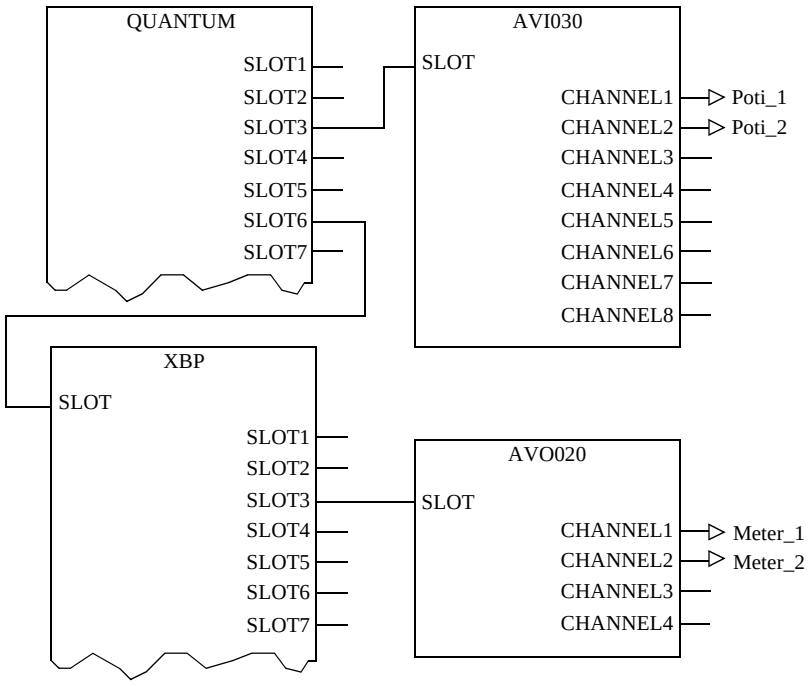
Procedure for expansion of the local backplane using XBE modules (Quantum)

Place a single QUANTUM EFB in the configuration section (CfgAnaIo).
The configuration data for the backplane expander is accessed using XBP EFBs.
Connect the XBP EFB to the XBE module slot.

Note: Do not specify Literals at the SLOT inputs of the configuration EFBs. SLOT inputs must be connected to SLOT outputs.

Example of expansion of the local backplane using XBE modules

Example of a configuration section "CfgAnaIo"



The EFB's mode of functioning can be found in the table below.

EFB	Function mode
Quantum	The EFB is used to edit the configuration data of a primary subrack for transfer by the analog in/out EFBs.
XBP	The EFB is used to prepare the backplane expander configuration data for transfer by the analog in/out EFBs.
AVI030	See *Example of only local I/O, p. 12*
AVO020	See *Example of only local I/O, p. 12*

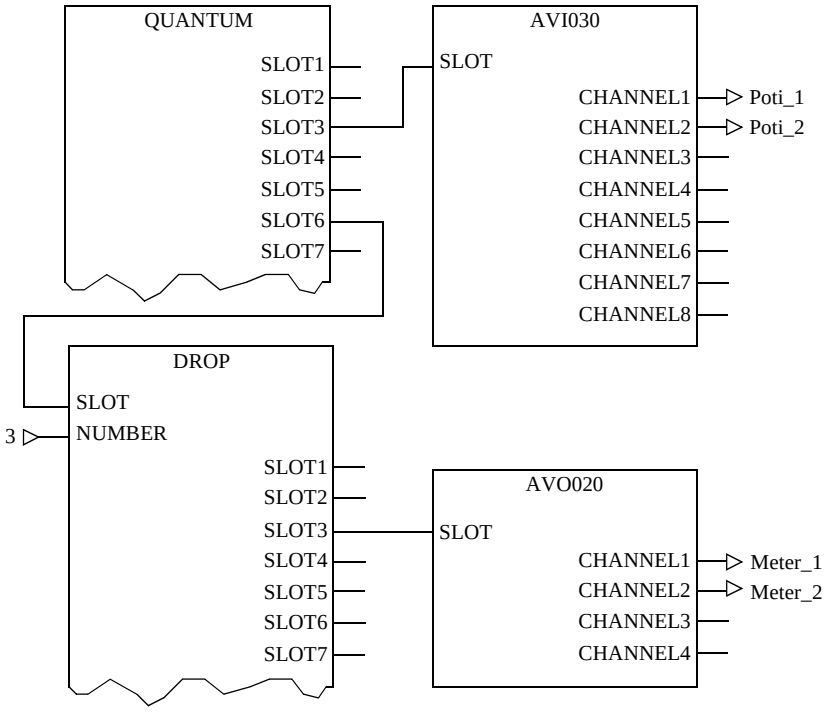
Procedure for distributed I/O (RIO, DIO) (Quantum)

Place a single QUANTUM EFB in the configuration section (CfgAnaIo).
The configuration data from remote I/O (RIO) and distributed I/O (DIO) or network option I/O (NOM) is accessed via DROP EFBs. The DROP EFB is applicable to all three I/O station types. If RIO is used, connect the DROP EFB to the slot on the RIO communication module. If DIO is used, connect the DROP EFB to the slot on the CPU or the NOM module. Each I/O station has its own address. Specify this number at the NUMBER input of the DROP EFB.

Note: Do not specify Literals at the SLOT inputs of the configuration EFBs. SLOT inputs must be connected to SLOT outputs.

Example of distributed I/O (RIO, DIO or NOM)

Example of a configuration section "CfgAnaIo"



The EFB's mode of functioning can be found in the table below.

EFB	Function mode
Quantum	The EFB is used to edit the configuration data of a primary subrack for transfer by the analog in/out EFBs.
Drop	The EFB is used to edit the configuration data of an I/O subrack for transfer by the analog in/out EFBs.
AVI030	See *Example of only local I/O, p. 12*
AVO020	See *Example of only local I/O, p. 12*

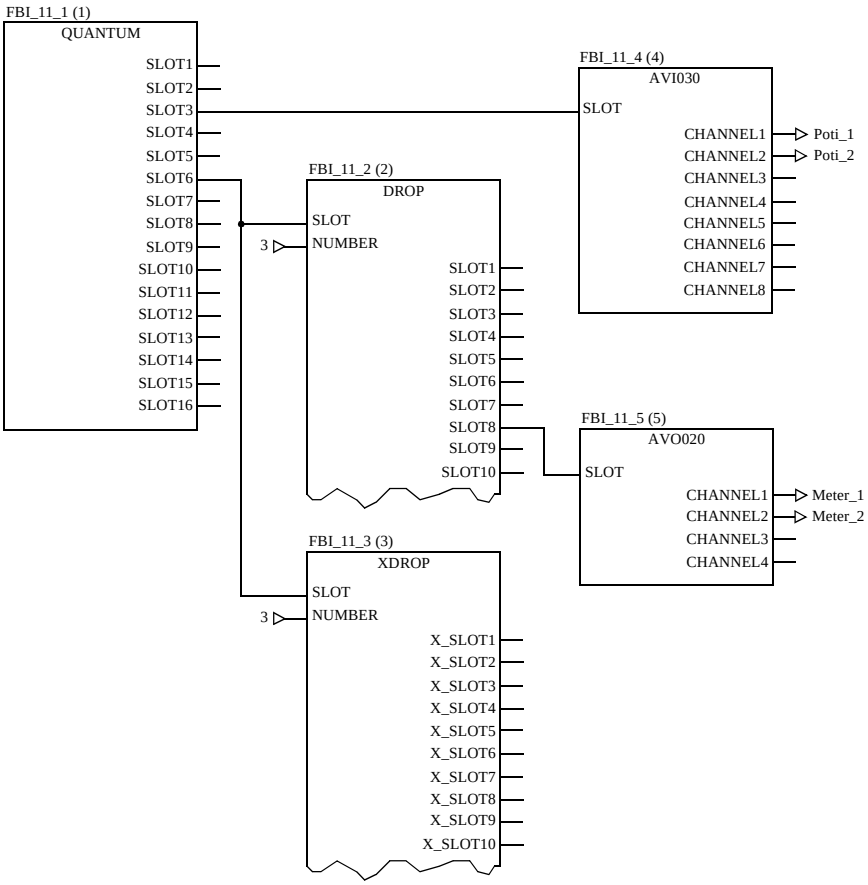
Procedure for expansion of the distributed backplane using XBE modules (Quantum)

Place a single QUANTUM EFB in the configuration section (CfgAnaIo).
The configuration data from remote I/O (RIO), distributed I/O (DIO) or network option I/O (NOM) is accessed via DROP EFBs. The DROP EFB is applicable to all three I/O station types. If RIO or NOM is used, connect the DROP EFB to the slot on the RIO communication module or the NOM module. If DIO is used, connect the DROP EFB to the slot on the CPU. Each I/O station has its own number. Specify this number at the NUMBER input of the DROP EFB.
The configuration data for the distributed backplane expander is accessed using XDROP EFBs. Connect the SLOT input of the XDROP EFB with the SLOT input of the DROP EFB. Enter the **same** number for the input NUMBER of the XDROP EFB as for the input NUMBER of the DROP EFB.

Note: Do not specify Literals at the SLOT inputs of the configuration EFBs. SLOT inputs must be connected to SLOT outputs.

Example of expansion of the distributed backplane using XBE modules

Example of a configuration section "CfgAnaIo"



The EFB's mode of functioning can be found in the table below.

EFB	Function mode
Quantum	The EFB is used to edit the configuration data of a primary subrack for transfer by the analog in/out EFBs.
XDROP	The EFB is used to prepare the distributed backplane expander configuration data for transfer by the analog in/out EFBs.
AVI030	See *Example of only local I/O, p. 12*
AVO020	See *Example of only local I/O, p. 12*

Scaling EFBs

At a Glance

The analog values are scaled using the EFBs of the "Analog IO Scaling" block library group in the scaling sections.
The analog I/O EFBs operate hardware-independent with the data types ANL_IN and ANL_OUT.

Scaling EFBs available

The following scaling EFBs are available:

- I_RAW, O_RAW:
  Raw value, no scaling
- I_RAWSIM:
  Raw value, simulation
- I_NORM, I_NORM_WARN, O_NORM, O_NORM_WARN:
  Standardization, representation in a range from 0.0 to 1.0
- I_PHYS, I_PHYS_WARN, O_PHYS, O_PHYS_WARN:
  Physical, physical range
- I_SCALE, I_SCALE_WARN, O_SCALE, O_SCALE_WARN:
  Scaled, representation in a user-defined range from MN to MX

Handy hints

Please take note of the following hints about using scaling EFBs:

- When using these EFBs the messages in **Online** → **Event viewer** should definitely be observed. That is where execution errors for these EFBs are recorded.
- For tasks not requiring the physical units and/or only scaling by 0-100% the NORM-EFBs, PHYS- or SCALE-EFBs are preferred.
- If scaling is required PHYS-EFBs should be used. PHYS-EFBs do not work without information about the physical units however, for these the SCALE-EFBs are used.
- SCALE-EFBs can also be used if physical scaling is not required or is not possible.
- SCALE-EFBs do not work in conjunction with input/output modules which provide direct physical values (e.g. decimal values). This applies, for example, to temperature or resistance modules not set to raw values.
- EFBs with "WARN" can not be used for all input/output modules. The descriptions of the individual EFBs explains which modules they can be used with.
- "Open circuit" is categorized as an error rather than a warning. "Open circuit" generates an online error message which can be accessed using **Online** → **Event viewer** and sets the output "ENO" of the "WARN"-EFBs to "0".
- The RAW-EFB is not usually required. It merely represents a simple way to make additional use of the raw values.

Debugging-EFBs

At a Glance

The data types ANL_IN and ANL_OUT are monitored by the EFBs of the "Analog I/O Debug" block library group in the scaling sections.
Ordinarily, the debug EFBs are not necessary. Although they can be used by system specialists for input/output diagnostics, e.g. to monitor the raw values in State RAM.

Scaling EFBs available

The following debug EFBs are available:

- I_DBSET,
- I_DEBUG,
- O_DBSET,
- O_DEBUG

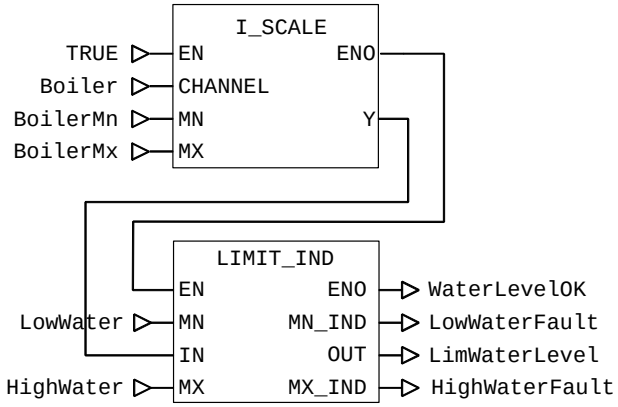
Application example for Quantum

At a Glance

To precisely monitor the output values, it is advisable to implement the scaling with two EFBs. The first EFB (scaling EFBs) scales the analog value and the second EFB monitors the scaled value for ranges preset by the process. In the following process, either the original Y output of the scaling EFB or the limited OUT output of the Limiter EFB can be used.

Application example

A simple example shows how the EFBs can be used.
The example assumes a boiler with a capacity of 350 liters. The input voltage ranges from 0.0 Volt for 0 liters to 10.0 Volt for 1000 liters. A PI controller should guarantee a volume between 200 and 300 liters. The Limiter EFB detects violations in this range and will limit the output.
Given values:
BoilerMn: 0
BoilerMx: 1 000
LowWater: 199
HighWater: 301
"Boiler" is an unlocated variable of the ANL_IN type and is linked to an AVI030-EFB.
Application example



Operating INTERBUS with Compact

3

Diagram of the INTERBUS structure with EFBs

Introduction

The IBS group in the ANA_IO block library contains all the Function blocks required for operating INTERBUS with TSX Compact for Concept PLC programming software. If the hardware is reconfigured (modules added or removed), the same modifications must be made to the PLC program as to the hardware. Modification of the addresses of all following modules is not necessary.

Note: The modules of the IBS group can also be set with the NOA 61100 and NOA 611 10.

Structure of an INTERBUS line

INTERBUS is constructed as a linear structure. This linear structure begins with the master and ends with an INTERBUS module. The diagram illustrates the INTERBUS hardware structure.
Hardware structure

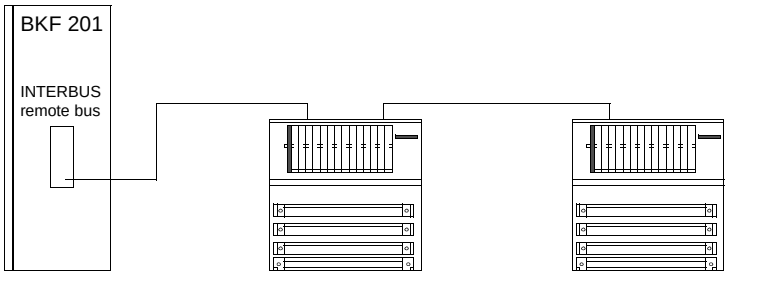
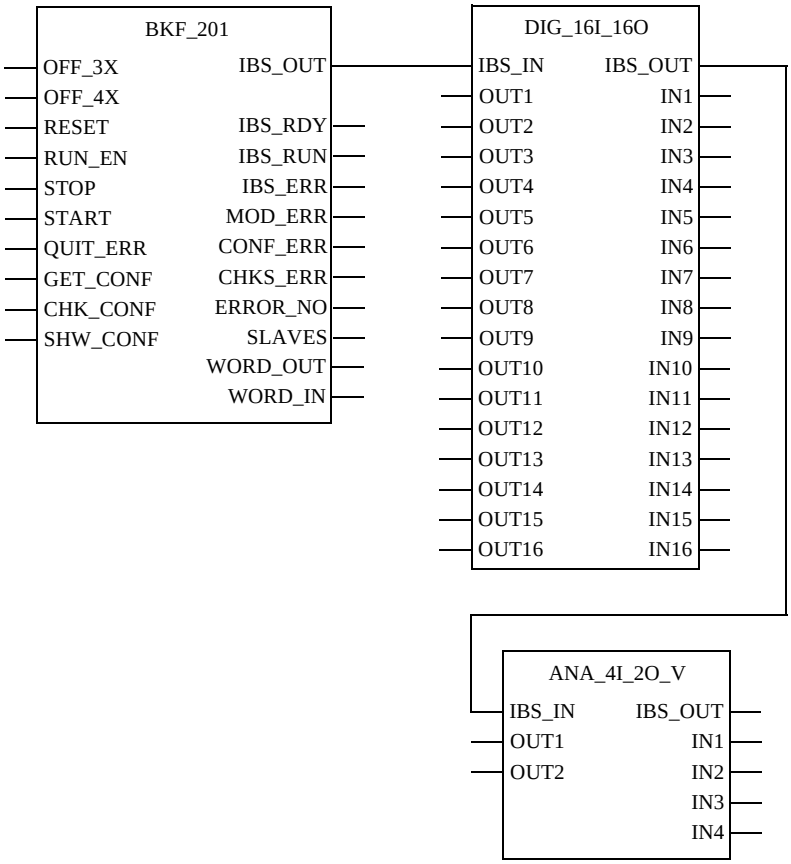


Diagram of the INTERBUS structure with EFBs

The diagram illustrates the same structure with the function blocks from the INTERBUS library.

Software structure with function blocks



Behavior of function blocks in simulator operation

IBS group function blocks in the ANA_IO library behave in the same way in simulator operation as with hardware connected. It should be noted, however, that the function block outputs labeled IN can only be assigned values if the outputs are assigned to a variable. A value can be assigned to the variables in the reference data editor. With hardware connected, no values can be assigned to the variables on the function block outputs.

Analog value processing with Momentum

4

At a Glance

Basic principle

Analog input data of I/O units from the Momentum product family are placed onto addresses with 3x references in State RAM. These are still raw values, although they were already linearized by the firmware of the I/O units. To make them available in the REAL format as voltage, current or temperature values, it is still necessary to first convert them in the user program. The required algorithms for this are in the user's guide "I/O Units for TSX Momentum".
The same procedure in reverse must be followed for the output values.
This conversion must always be done, regardless of the form in which data reaches State RAM (via Modbus Plus PeerCop, I/O Bus, ...).
For easier handling, Concept offers EFBs that perform this conversion.

Affected modules

This includes all analog I/O units and TIO modules.

What's in this chapter?

This chapter contains the following topics:

Procedure for Analog value processing with Momentum

Concept EFBs

To calculate analog input or output data for Momentum special concept EFBs exist. The following table can be used to decide which EFB is necessary for which module:

Module	EFB
170 AAI 140 00	ANA_16I (See *ANA_16I: Configuring the module AAI 140 00, p. 81*)
170 AAI 520 40	ANA_4I_M (See *ANA_4I_M: Configuring the module AAI 520 40, p. 87*)
170 AAO 120 00	ANA_4O (See *ANA_4O: Configuring the module BAO 126 00, p. 113*)
170 AAO 921 00	ANA_4O (See *ANA_4O: Configuring the module BAO 126 00, p. 113*)
170 AMM 090 00	MIX_4I_2O (See *MIX_4I_2O: Configuring the AMM module 090 00, p. 247*)
170 BAI 036 00	ANA_8I (See *ANA_8I: Configuring the module AAI 030 00, BAI 036 00, p. 119*)
170 BAM 096 00	ANA_4I_2O (See *ANA_4I_2O: Configuring the TIO-module BAM 096 00, p. 95*) ANA_4I_2O_C (See *ANA_4I_2O_C: Configuring the TIO-module BAM 096 00, p. 105*) (current) ANA_4I_2O_V (See *ANA_4I_2O_V: Configuring the TIO-module BAM 096 00, p. 109*) (voltage)
170 BAO 126 00	ANA_4O (See *ANA_4O: Configuring the module BAO 126 00, p. 113*)

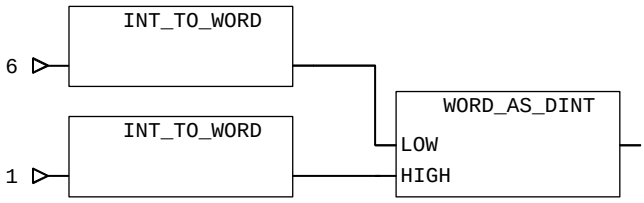
Procedure

Procedure for converting analog values

Step	Action
1	Incorporate the EFB that matches your module into your program.
2	Assign the parameters for the channels of your modules to the pins PM_OUT ... and PM_IN... (enter decimal values). The codes for the parameters are given in the user's guide "I/O Units for TSX Momentum". They must be consistent with the parameter assignment in the I/O map.
3	Assign the module address to the IBS_IN pin (data type DINT) (address in State RAM after which module data is given). The high word contains the offset of the first 3x address of the module, the low word the offset of the first 4x address. The parameter addresses have to be considered as well.

Examples

Example 1:
The following logic simplifies the value determination for IBS_IN, in the example here, a 170 AAI 140 00 with the addresses 300 001 .... 300 016 and 400 006 .... 400 021. For the AAI 140, the EFB ANA_16I (See *ANA_16I: Configuring the module AAI 140 00, p. 81*) will always occupy 16 output words! Consequently, enter the offset for the outputs on top and for the inputs on the bottom.
If your network has several I/O units, the storage of this logic in a DFB is recommended.
DFB logic



Example 2:
If using Peer Cop and placing the measuring values of a 170 AAI 140 onto the 300 001 .... 300 116 addresses, the same values can be entered with the consideration that the EFB does not only occupy the addresses 400 006 ... 400 009 for the I/O unit parameters, but internally also the addresses 400 010 ... 400 021.
Example 3:
If it is necessary to convert REAL values into raw values and have these on the addresses 400 101 …. 400 104 in order to place them onto the outputs of a 170 AAO 120, for instance via Peer Cop, the value entered on top must be 100 and the value on the bottom must be 0 (the EFB for the AAO does not occupy any input words). Please note, the EFB has to have the address 400 100 for the I/O unit parameters.

Result of the described procedure

The correct values of the modules are now present as data type REAL. They can be scaled, can be used for calculations or closed-loop control.

Note: Do not forget to divide the values by 4 when working with voltages. (Please refer to the EFB documentation as well).

Example analog value processing with Momentum

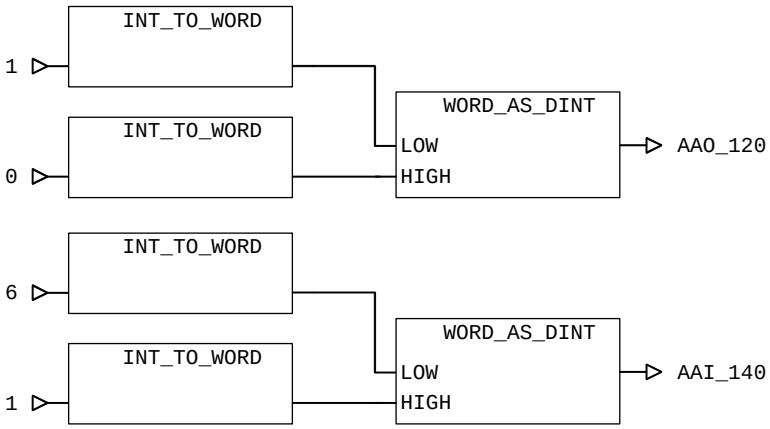
Task

There are two analog modules on the I/O bus:

- a 170 AAO 120 00, addressed to 400 001 400.005, disconnect behavior: all outputs to 0
- a 170 AAI 140 00, addressed to 300 001 300 016 and 400 006 400 021, channels 1 ... 4 are parametered for 4 … 20 mA, all other channels are disconnected.

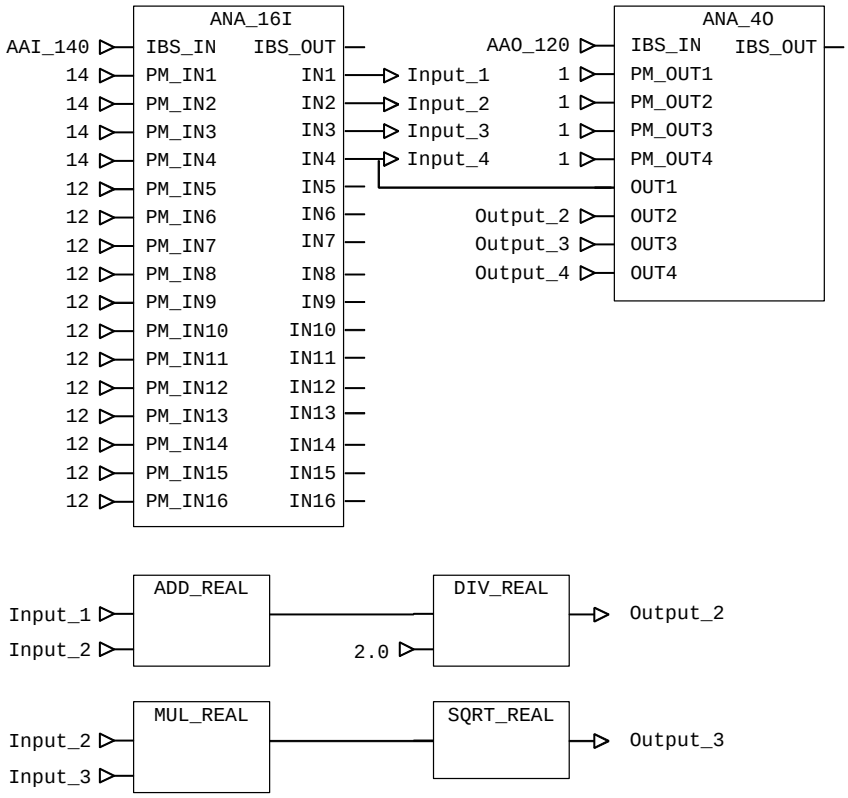
Addressing

The following network can be used for the addressing:



Network with analog value EFBs

The raw values in this example are in the 3x and 4x registers, the converted REAL values for the inputs are carried in the unlocated variables Input_1…Input_4 and for the outputs in Output_1 …Output_4.
The network with the analog value EFBs will look as follows:



Also note, the values can be easily used to continue computing or to be placed again directly onto an EFB for an output module.
If the module addresses are continuous and have no gaps, the pins IBS_OUT and IBS_IN from EFBs of two consecutive modules can be connected while only the first address has to be entered.

Note for digital Momentum I/O units

Concept also makes EFBs available for digital I/O units and modules. They convert the word references to bit references (and vice versa).
The following EFBs are available:

Module	EFB
170 ADI 340 00	DIG_16I (See *DIG_16I: Configuring the TIO-module BDI 346 00 / 546 50 / 746 50, p. 169*)
170 ADI 350 00	DIG_16I (See *DIG_16I: Configuring the TIO-module BDI 346 00 / 546 50 / 746 50, p. 169*) (2 EFBs connected in series, IBS_OUT connected with IBS_IN)
170 ADI 540 00	DIG_16I (See *DIG_16I: Configuring the TIO-module BDI 346 00 / 546 50 / 746 50, p. 169*)
170 ADM 350 x0	DIG_16I_16O (See *DIG_16I_16O: Configuring the TIO-module BDM 346 00, p. 181*)
170 ADM 370 10	DIG_16I_16O (See *DIG_16I_16O: Configuring the TIO-module BDM 346 00, p. 181*)
170 ADM 390 10	DIG_16I_12O_MON (See *DIG_16I_12O_MON: Configuring the module ADM 390 10, p. 175*)
170 ADM 390 30	DIG_16I_16O (See *DIG_16I_16O: Configuring the TIO-module BDM 346 00, p. 181*)
170 ADM 690 50	DIG_16I_16O (See *DIG_16I_16O: Configuring the TIO-module BDM 346 00, p. 181*)
170 ADO 340 00	DIG_16O (See *DIG_16I_16O: Configuring the TIO-module BDM 346 00, p. 181*)
170 ADO 350 00	DIG_16I (See *DIG_16I_16O: Configuring the TIO-module BDM 346 00, p. 181*) (2 EFBs connected in series, IBS_OUT connected with IBS_IN)
170 ADO 5x0 50	DIG_16O (See *DIG_16I_16O: Configuring the TIO-module BDM 346 00, p. 181*)
170 ADO 7x0 50	DIG_16O (See *DIG_16I_16O: Configuring the TIO-module BDM 346 00, p. 181*)

Note: Modules that have the same Ident code can use the same EFB.

EFB descriptions

II

Overview

At a Glance

These EFB descriptions are listed in alphabetical order.

What's in this part?

This part contains the following chapters:

ACI030: Configuring the Quantum module ACI 030 00

5

Overview

At a Glance

This chapter describes the block ACI030.

What's in this chapter?

This chapter contains the following topics:

Brief description

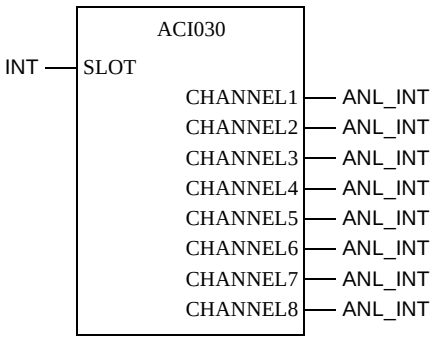
Function description

The Function block is used to edit the configuration data of an ACI 030 00 Quantum module for subsequent use by the scaling EFBs.
This module has 8 unipolar input channels for mixed voltage and current processing.
For the configuration of an ACI 030 00 the function block in the configuration section is connected to the corresponding slot output of the QUANTUM or DROP Function block. The 3x references specified in the I/O map are automatically assigned internally to the individual channels and can therefore only be occupied by Unlocated ariables.
The analog values can be further processed in Scaling Sections using the I_DEBUG, I_NORM, I_RAW and I_SCALE Function blocks.
EN and ENO can be projected as additional parameters.

Representation

Symbol

Block representation:



Parameter description

Block parameter description:

Parameter	Data type	Meaning
SLOT	INT	Slot
CHANNEL1	ANL_IN	Channel 1
CHANNEL2	ANL_IN	Channel 2
CHANNEL3	ANL_IN	Channel 3
CHANNEL4	ANL_IN	Channel 4
CHANNEL5	ANL_IN	Channel 5
CHANNEL6	ANL_IN	Channel 6
CHANNEL7	ANL_IN	Channel 7
CHANNEL8	ANL_IN	Channel 8

Runtime error

Runtime error

If no ACI 030 00 module has been configured for the specified SLOT input, an error message appears.
The status information "Open circuit or undervoltage on channel" can be collected via the status register defined in the I/O map.

ACI040: Configuring the Quantum module ACI 040 00

6

Overview

At a Glance

This chapter describes the block ACI040.

What's in this chapter?

This chapter contains the following topics:

Brief description

Function description

The Function block is used to edit the configuration data of the ACI 040 00 Quantum module for subsequent use by the scaling EFBs.

The module has 16 channels which, according to requirement, can be used as differential or single inputs for the processing of current. When processing current the ranges are 0…20 mA, 0…25 mA and 4…20 mA.

For the configuration of an ACI 040 00 the function block in the configuration section is connected to the corresponding SLOT output of the QUANTUM or DROP Function block. The 3x references specified in the I/O map are assigned internally to individual channels automatically. The channels can only be occupied by Unlocated variables.

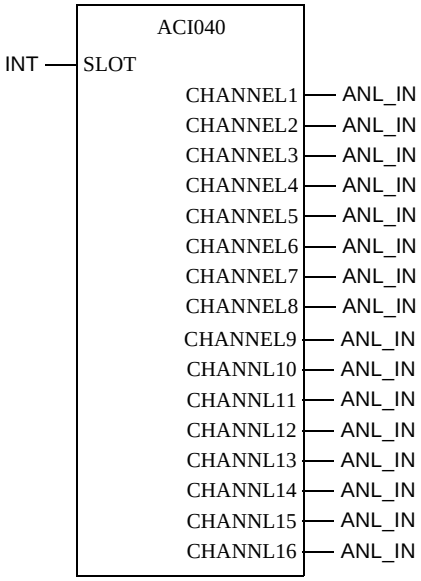
The analog values can be processed further in Scaling Sections using the I_NORM, I_PHYS, I_RAW and I_SCALE Function blocks. I_DEBUG and I_DBSET are also available for testing purposes.

EN and ENO can be projected as additional parameters.

Representation

Symbol

Block representation:



Parameter description

Block parameter description:

Parameter	Data type	Meaning
SLOT	INT	Slot
CHANNEL1	ANL_IN	Channel 1
:	:	:
CHANNEL9	ANL_IN	Channel 9
CHANNL10	ANL_IN	Channel 10
:	:	:
CHANNL16	ANL_IN	Channel 16

Runtime error

Runtime error

If no ACI 040 00 module has been configured for the specified SLOT input, an error message appears.

In the "4 to 20 mA" mode, the "open circuit on channel" status information is available. It can be collected via the 3 references of the module (3x+16) defined in the I/O mapping.

ACO020: Configuring the Quantum module ACO 020 00

7

Overview

At a Glance

This chapter describes the block ACO20.

What's in this chapter?

This chapter contains the following topics:

Brief description

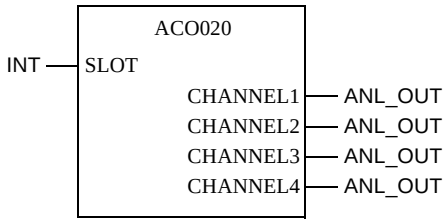
Function description

The Function block is used to edit the configuration data of an ACO 020 00 Quantum module for subsequent use by the scaling EFBs.
This module has 4 output channels for the processing of current in the range of 4 ... 20 mA.
For the configuration of an ACO 020 the function block in the configuration section is connected to the corresponding slot output of the QUANTUM or DROP Function block. The 4x references specified in the I/O map and the status information (if configured) are automatically assigned internally to the individual channels and can therefore only be occupied by Unlocated variables.
The analog values can be further processed in Scaling sections using the O_DEBUG, O_NORM, O_RAW and O_SCALE function blocks. The parameters EN and ENO can additionally be projected.

Representation

Symbol

Block representation:



Parameter description

Block parameter description:

Parameter	Data type	Meaning
SLOT	INT	Slot
CHANNEL1	ANL_OUT	Channel 1
CHANNEL2	ANL_OUT	Channel 2
CHANNEL3	ANL_OUT	Channel 3
CHANNEL4	ANL_OUT	Channel 4

Runtime error

Runtime error

If no ACO 020 00 module has been configured for the specified SLOT input, an error message appears.
The status information "Open circuit on channel" can be collected via the status register defined in the I/O map.

ACO130: Configuring the Quantum module ACO 130 00

8

Overview

At a Glance

This chapter describes the block ACO130.

What's in this chapter?

This chapter contains the following topics:

Brief description

Function description

The Function block is used to edit the configuration data of an ACO 130 00 Quantum module for subsequent use by the scaling EFBs.

This module has 8 output channels for controlling and supervising the currents in the ranges 0 to 20 mA, 0 to 25 mA and 4 to 20 mA.

For the configuration of an ACO 130 00 the function block in the configuration section is connected to the corresponding SLOT output of the QUANTUM or DROP Function block. The 4x referencesspecified in the I/O map are automatically assigned internally to individual channels. The channels can only be occupied byUnlocated variables.

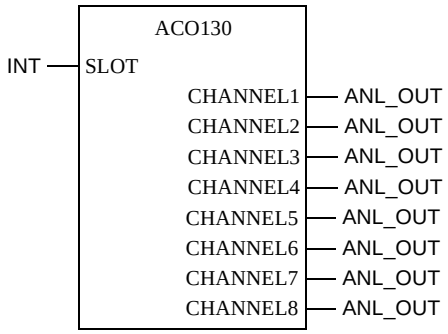
The analog values can be further processed in the Scaling Sections using the O_NORM, O_PHYS, O_RAW and O_SCALE Function blocks. O_DEBUG and O_DBSET are also available for testing purposes.

EN and ENO can be projected as additional parameters.

Representation

Symbol

Block representation:



Parameter description

Block parameter description:

Parameter	Data type	Meaning
SLOT	INT	Slot
CHANNEL1	ANL_OUT	Channel 1
:	:	:
CHANNEL8	ANL_OUT	Channel 8

Runtime error

Runtime error

If no ACO 130 00 module has been configured for the specified SLOT input, an error message will appear.

In the "4 to 20 mA" mode, the "open circuit on channel" status information is available. It can be collected via the status register defined in the I/O mapping.

ADU204: Configuring the Compact module ADU 204

9

Overview

At a Glance

This chapter describes the block ADU204.

What's in this chapter?

This chapter contains the following topics:

Brief description

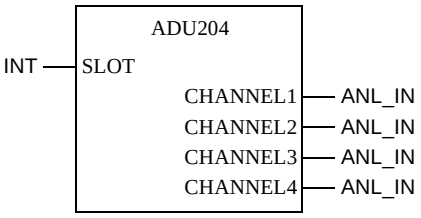
Function description

The Function block is used to edit the configuration data of a ADU 204 Compact module for subsequent use by the scaling EFBs.
This module has four input channels for measuring temperature and resistance.
For the configuration of an ADU 204, the function block in the Configuration section is connected to the corresponding SLOT output of the COMPACT Function block.
The 3x references specified in the I/O map are automatically assigned internally to the individual channels and can therefore only be occupied by Unlocated Variables.
The analog values can be further processed in Scaling sections using the I_DEBUG, I_NORM, I_SCALE, I_PHYS, I_DBSET, I_RAW Function blocks.
EN and ENO can be configured as additional parameters.

Representation

Symbol

Block representation:



Parameter description

Block parameter description:

Parameter	Data type	Meaning
SLOT	INT	Slot
CHANNEL1	ANL_IN	Channel 1
CHANNEL2	ANL_IN	Channel 2
CHANNEL3	ANL_IN	Channel 3
CHANNEL4	ANL_IN	Channel 4

Runtime error

Runtime error

If no ADU 204 module has been configured for the specified SLOT input, an error message appears.
The status information "Range violation" on a channel can be collected via the status entry in the ANL_IN data structure.

ADU205: Configuring the Compact module ADU 205

10

Overview

At a Glance

This chapter describes the block ADU205.

What's in this chapter?

This chapter contains the following topics:

Brief description

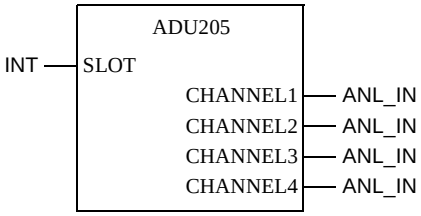
Function description

The Function block is used to edit the configuration data of a ADU 205 Compact module for subsequent use by the scaling EFBs.
This module has four input channels for combined bipolar and unipolar voltage and current processing.
For the configuration of an ADU 205, the function block in the Configuration section is connected to the corresponding SLOT output of the COMPACT Function block.
The 3x references specified in the I/O map are automatically assigned internally to the individual channels and can therefore only be occupied by Unlocated Variables.
The analog values can be further processed in Scaling sections using the I_DEBUG, I_NORM, I_SCALE, I_PHYS, I_DBSET, I_RAW Function blocks.
EN and ENO can be configured as additional parameters.

Representation

Symbol

Block representation:



Parameter description

Block parameter description:

Parameter	Data type	Meaning
SLOT	INT	Slot
CHANNEL1	ANL_IN	Channel 1
CHANNEL2	ANL_IN	Channel 2
CHANNEL3	ANL_IN	Channel 3
CHANNEL4	ANL_IN	Channel 4

Runtime error

Runtime error

If no ADU 205 module has been configured for the specified SLOT input, an error message appears.
A range warning for the channels is evaluated by the processing Function blocks (See *Function description, p. 54*).
The status information "Range violation" or "Open circuit" on a channel can be collected via the status entry in the ANL_IN data structure.

ADU206: Configuring the Compact module ADU 206/ ADU 256

11

Overview

At a Glance

This chapter describes the block ADU206.

What's in this chapter?

This chapter contains the following topics:

Brief description

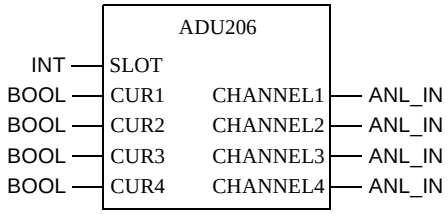
Function description

The Function block is used to edit the configuration data of an ADU 206 / ADU256 Compact module for subsequent use by the scaling EFBs.
This module has four input channels for combined bipolar and unipolar voltage and current processing.
For the configuration of an ADU 206 / ADU 256 the function block in the configuration section
is connected to the corresponding output of the function block COMPACT. The 3x references specified in the I/O map are automatically assigned internally to the individual channels and can therefore only be occupied by Unlocated Variables.
The CURx parameters indicate whether a channel is configured to process current (TRUE) or voltage (FALSE). As this card has range settings (in the parametering of this card in the I/O map) which do not allow current processing, the relevant CURx input must in this case be FALSE.
The analog values can be further processed in Scaling sections using the I_DEBUG, I_NORM, I_SCALE, I_PHYS, I_DBSET, I_RAW Function blocks.
EN and ENO can be configured as additional parameters.

Representation

Symbol

Block representation:



Parameter description

Block parameter description:

Parameter	Data type	Meaning
SLOT	INT	Slot
CUR1	BOOL	0: Channel 1 processing voltage 1: Channel 1 processing current
CUR2	BOOL	0: Channel 2 processing voltage 1: Channel 2 processing current
CUR3	BOOL	0: Channel 3 processing voltage 1: Channel 3 processing current
CUR4	BOOL	0: Channel 4 processing voltage 1: Channel 4 processing current
CHANNEL1	ANL_IN	Channel 1
CHANNEL2	ANL_IN	Channel 2
CHANNEL3	ANL_IN	Channel 3
CHANNEL4	ANL_IN	Channel 4

Runtime error

Runtime error

If no ADU 206 / ADU 256 module has been configured for the specified SLOT input, an error message will appear.
If current processing is selected (TRUE at the relevant CURx input) for a channel, which is only configured to allow voltage processing, an error message appears with the number of the affected channel (1-4).
A range warning for the channels is evaluated by the processing Function blocks (See *Function description, p. 58*).
The status information "Open circuit or range violation on channel" can be collected via the status register (3x reference) defined in the I/O map.3

ADU214: Configuring the Compact module ADU 214

12

Overview

At a Glance

This chapter describes the block ADU214.

What's in this chapter?

This chapter contains the following topics:

Brief description

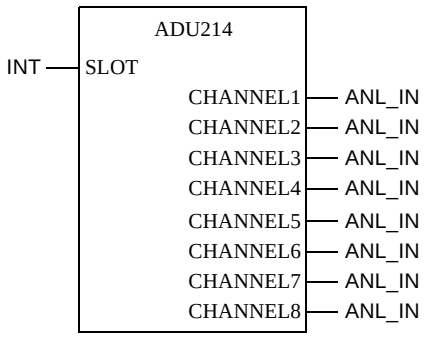
Function description

The Function block is used to edit the configuration data of a ADU 214 Compact module for subsequent use by the scaling EFBs.
The module has a maximun of 8 analog input channels for voltage, current, resistance and temperature measurement. They can be used as unipole or as bipolar inputs. Mixed operation is also allowed.
For the configuration of an ADU 214, the function block in the Configuration section is connected to the corresponding SLOT output of the COMPACT Function block.
The 3x references specified in the I/O map are automatically assigned internally to the individual channels and can therefore only be occupied by Unlocated variables.
The analog values can be further processed in Scaling sections using the I_DEBUG, I_NORM, I_SCALE, I_PHYS, I_DBSET, I_RAW Function blocks.
EN and ENO can be configured as additional parameters.

Representation

Symbol

Block representation:



Parameter description

Block parameter description:

Parameter	Data type	Meaning
SLOT	INT	Module slot
CHANNEL1	ANL_IN	Channel 1
CHANNEL2	ANL_IN	Channel 2
CHANNEL3	ANL_IN	Channel 3
CHANNEL4	ANL_IN	Channel 4
CHANNEL5	ANL_IN	Channel 5
CHANNEL6	ANL_IN	Channel 6
CHANNEL7	ANL_IN	Channel 7
CHANNEL8	ANL_IN	Channel 8

Runtime error

Runtime error

If no ADU 214 module has been configured for the specified SLOT input, an error message appears.
The error information for a channel, e.g. "line break or range exceeded" can be collected via the status register defined in the I/O map. If errors occur simultaneously on several channels, the error of the lowest channel number will be shown until it is removed. Then the error message for the next higher channel number is given, etc.

AII330: Configuring the Quantum module AII 330 00

13

Overview

At a Glance

This chapter describes the block AII330.

What's in this chapter?

This chapter contains the following topics:

Brief description

Function description

The Function block is used to edit the configuration data of an AII 330 00 Quantum module for subsequent use by the scaling EFBs.

The module has 8 intrinsically safe channels and can be used either as a resistor temperature sensor (RTD) or as a thermoelement/millivolt input module.

For the configuration of an AII 330 00 the function block in the configuration section is connected to the corresponding SLOT output of the QUANTUM or DROP Function block. The 3x referencesspecified in the I/O map are automatically assigned internally to individual channels. The channels can only be occupied byUnlocated variables.

The analog values can be further processed inScaling sections using the I_NORM, I_NORM_WARN, I_PHYS, I_PHYS_WARN, I_RAW, I_SCALE and I_SCALE_WARN Function blocks. I_DEBUG and I_DBSET are also available for testing purposes.

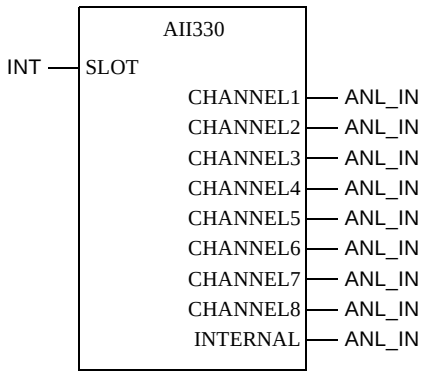
Note: I_SCALE and I_SCALE_WARN can not be used to set parameters for physical or temperature values.

EN and ENO can be configured as additional parameters.

Representation

Symbol

Block representation:



Parameter description

Block parameter description:

Parameters	Data type	Meaning
SLOT	INT	Module slot
CHANNEL1	ANL_IN	Channel 1
CHANNEL2	ANL_IN	Channel 2
CHANNEL3	ANL_IN	Channel 3
CHANNEL4	ANL_IN	Channel 4
CHANNEL5	ANL_IN	Channel 5
CHANNEL6	ANL_IN	Channel 6
CHANNEL7	ANL_IN	Channel 7
CHANNEL8	ANL_IN	Channel 8
INTERNAL	ANL_IN	Temperature of module

Runtime error

Runtime error

If no AII 330 00 module has been configured for the specified SLOT input, an error message will appear.

The range warning for the channels can be evaluated using the function block I_NORM_WARN, I_SCALE_WARN or I_PHYS_WARN.

The status information "Open circuit or range violation on channel" can be collected via the 3 references (3x-8; module status register) defined in the I/O map or via the status register defined in the I/O map. (The information in the status register is a copy of the 3x+8 module status register (High-Byte).

AII 330 10: Configuring the Quantum module AII 330 10

14

Overview

At a Glance

This chapter describes the block AII 330 10.

What's in this chapter?

This chapter contains the following topics:

Brief description

Function description

The Function block is used to edit the configuration data of an AII 330 10 Quantum module for subsequent use by the scaling EFBs.

The module has 8 unipolar intrinsically safe channels. The following ranges can be selected: 0 to 20 mA , 0 to 25 mA and 4 to 20 mA.

For the configuration of an AII 330 00 the function block in the configuration section is connected to the corresponding SLOT output of the QUANTUM or DROP Function block. The 3x referencesspecified in the I/O map are automatically assigned internally to individual channels. The channels can only be occupied byUnlocated variables.

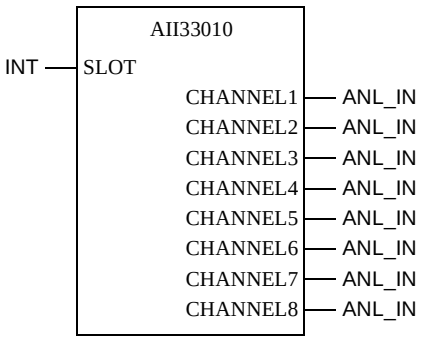
The analog values can be further processed in Scaling Sections using the I_NORM, I_PHYS, I_RAW and I_SCALE Function blocks. I_DEBUG and I_DBSET are also available for testing purposes.

EN and ENO can be projected as additional parameters.

Representation

Symbol

Block representation:



Parameter description

Block parameter description:

Parameter	Data type	Meaning
SLOT	INT	Slot
CHANNEL1	ANL_IN	Channel 1
CHANNEL2	ANL_IN	Channel 2
CHANNEL3	ANL_IN	Channel 3
CHANNEL4	ANL_IN	Channel 4
CHANNEL5	ANL_IN	Channel 5
CHANNEL6	ANL_IN	Channel 6
CHANNEL7	ANL_IN	Channel 7
CHANNEL8	ANL_IN	Channel 8

Runtime error

Runtime error

If no AII 330 10 module has been configured for the specified SLOT input, an error message will appear.

In the "4 to 20 mA" mode, the "open circuit on channel" status information is available. It can be collected via the status register defined in the I/O mapping.

AIO330: Configuring the Quantum module AIO 330 00

15

Overview

At a Glance

This chapter describes the block AIO330.

What's in this chapter?

This chapter contains the following topics:

Brief description

Function description

The Function block is used to edit the configuration data of an AIO 330 00 Quantum module for subsequent use by the scaling EFBs.

This module has 8 intrinsically safe symmetrical output channels for controlling and supervising the currents in the ranges 0 to 20 mA, 0 to 25 mA and 4 to 20 mA.

For the configuration of an AIO 330 00 the function block in the configuration section is connected to the corresponding SLOT output of the QUANTUM or DROP Function block. The 4x referencesspecified in the I/O map are automatically assigned internally to individual channels. The channels can only be occupied byUnlocated variables.

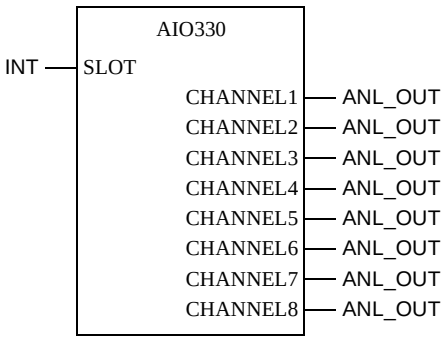
The analog values can be further processed in the Scaling Sections using the O_NORM, O_PHYS, O_RAW and O_SCALE Function blocks. O_DEBUG and O_DBSET are also available for testing purposes.

EN and ENO can be projected as additional parameters.

Representation

Symbol

Block representation:



Parameter description

Block parameter description:

Parameter	Data type	Meaning
SLOT	INT	Slot
CHANNEL1	ANL_OUT	Channel 1
CHANNEL2	ANL_OUT	Channel 2
CHANNEL3	ANL_OUT	Channel 3
CHANNEL4	ANL_OUT	Channel 4
CHANNEL5	ANL_OUT	Channel 5
CHANNEL6	ANL_OUT	Channel 6
CHANNEL7	ANL_OUT	Channel 7
CHANNEL8	ANL_OUT	Channel 8

Runtime error

Runtime error

If no AIO 330 00 module has been configured for the specified SLOT input, an error message will appear.

The status information "Open circuit or range violation on channel" can be collected via the status register defined in the I/O map.

AMM090: Configuring the Quantum module AMM 090

16

Overview

At a Glance

This chapter describes the block AMM090.

What's in this chapter?

This chapter contains the following topics:

Brief description

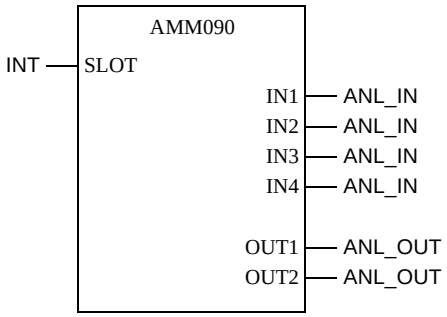
Function description

The Function block is used to edit the configuration data of an AMM 090 00 Quantum module for subsequent use by the scaling EFBs.
This module has 4 bipolar input channels for mixed voltage and current processing. The module also has 2 current output channels.
For the configuration of an AMM 090 00 the function block in the configuration section is connected to the corresponding slot output of the QUANTUM or DROP Function block. The 3x references and 4x references specified in the I/O map are automatically assigned internally to the individual channels and can therefore only be occupied by Unlocated variables.
The analog values can be further processed in Scalings sections using the function blocks I_DEBUG, I_NORM, I_NORM_WARN, I_PHYS, I_PHYS_WARN, I_RAW, I_SCALE, I_SCALE_WARN for the inputs and O_DEBUG, O_NORM, O_SCALE, O_PHYS, O_DBSET , O_RAW for the outputs.
EN and ENO can be configured as additional parameters.

Representation

Symbol

Block representation:



Parameter description

Block parameter description:

Parameter	Data type	Meaning
SLOT	INT	Slot
IN1	ANL_IN	Input channel 1
IN2	ANL_IN	Input channel 2
IN3	ANL_IN	Input channel 3
IN4	ANL_IN	Input channel 4
OUT1	ANL_OUT	Output channel 1
OUT2	ANL_OUT	Output channel 2

Runtime error

Runtime error

If no AMM 090 00 module has been configured for the specified SLOT input, an error message appears.
The range warning for the input channels can be evaluated using the I_NORM_WARN, I_PHYS_WARN or I_SCALE_WARN Function blocks.
The outputs return no warnings.
The status message "Open circuit or range violation on channel" can be requested via the status register defined in the I/O map.

ANA_16I: Configuring the module AAI 140 00

17

Overview

At a Glance

This chapter describes the block ANA_16I.

What's in this chapter?

This chapter contains the following topics:

Brief description

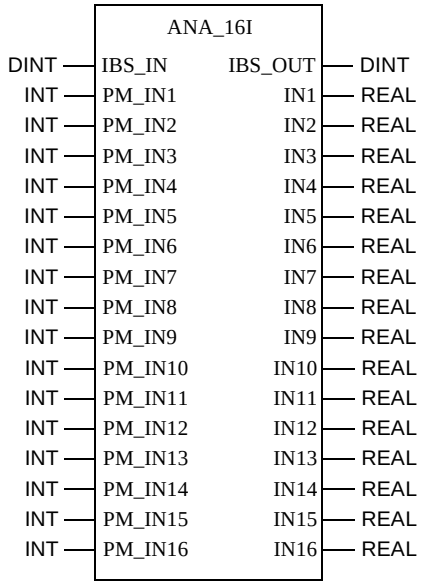
Function description

The Function block is a software connection to the INTERBUS Momentum/IS 170 AAI 140 00 hardware module.
The function block has 16 analog inputs. The function block must be parametered in the same way as the hardware module.
EN and ENO can be projected as additional parameters.

Representation

Symbol

Block representation:



Parameter description

Block parameter description:

Parameter	Data type	Meaning
IBS_IN	DINT	Incoming INTERBUS
PM_IN1	INT	Parameter for input 1
:	:	:
PM_INT	INT	Parameter for input 16
IBS_OUT	DINT	Outgoing INTERBUS
IN1	REAL	Input 1 of the module
:	:	:
IN16	REAL	Input 16 of the module

Detailed description

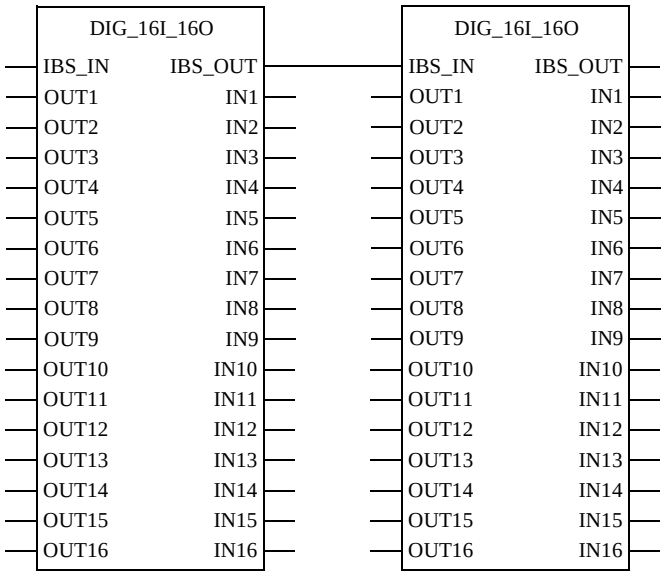
Detailed description

The funciton block occupies 16 input words and 16 outputwords in the Status-RAM.

Parameter description - inputs

IBS_IN

IBS_IN = Connection for the incoming remote bus part of INTERBUS
On the hardware, the male connector is on the top left of the module. Here, the module is connected to the outgoing remote bus (IBS_OUT) of the master (1st module on the bus) or the preceding module (see diagram). The link can be made via a line or via a variable. For the hardware, the type of connection corresponds to the INTERBUS cable between two bus devices.
Connection of two INTERBUS modules



PM_INx

PM_INx = Parameters for the input channels
x stands for the digit 1 to 16 which indicates the particular input channel.
These parameters are used to parameter the input channels.
The meaning of the parameter values can be found in the table below.

Parameter value	Meaning
0	reserved
10	+/- 5V
11	+/- 10V
12	Channel inactive
14	4...20 mA

Example:
Input 3 should be 4 ...20 mA.
PM_IN3 = "14"

Note: The reserved parameter codes are not accepted by the module, i.e. the last parameter used will still apply.

Parameter description - Outputs

IBS_OUT

IBS_OUT = Connection for the outgoing remote bus part of INTERBUS
On the hardware, the male connector is on the top right of the module. The module is connected to the incoming remote bus (IBS_IN) of the following module, either via a line or via a variable. For the hardware, the type of connection corresponds to the INTERBUS cable between two INTERBUS modules.

INx

INx = input channel x
x stands for the digit 1 to 16 which indicates the particular input channel. The analog process values are read into the INTERBUS module via the relevant input (INx).

Note: The values to be applied here are standardized, i.e. given as voltage in volts or as current in milliamperes. The input as current or voltage depends on the parameters set for the particular channel. .

ANA_4I_M: Configuring the module AAI 520 40

18

Overview

Introduction

This chapter describes the block ANA_41_M.

What's in this chapter?

This chapter contains the following topics:

Brief description

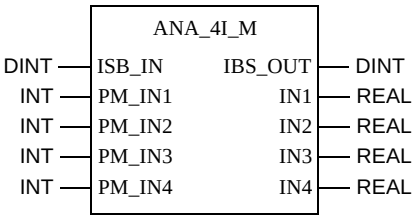
Function description

The ANA_4I_M Function block (M = measurement) is a software connection to the Momentum/IS 170 AAI 520 40 hardware module.
The function block has 4 analog inputs and is a special block for temperature and extra-low voltage measurements. The function block must be parametered in the same way as the hardware module.
EN and ENO can be projected as additional parameters.

Representation

Symbol

Block representation:



Parameter description

Block parameter description:

Parameter	Data type	Meaning
IBS_IN	DINT	Incoming INTERBUS
PM_IN1	INT	Parameter for input 1
:	:	:
PM_IN4	INT	Parameter for input 4
IBS_OUT	DINT	Outgoing INTERBUS
IN1	REAL	Input 1 of the module
:	:	:
IN4	REAL	Input 4 of the module

Detailed description

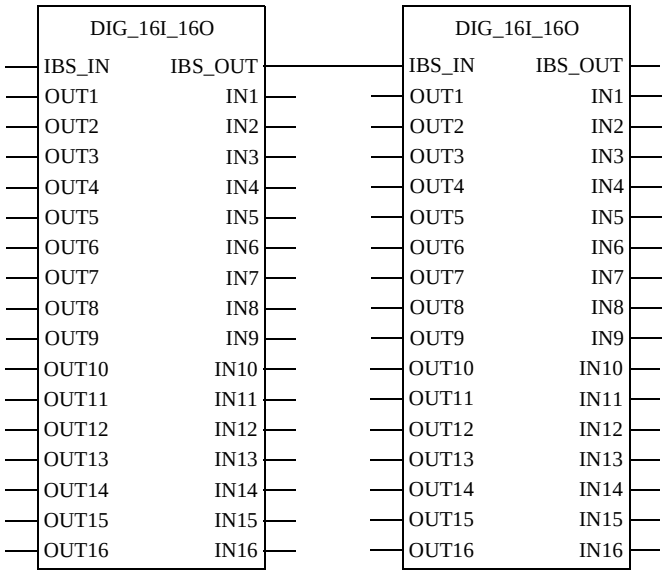
Detailed description

The block occupies 4 input words and 4 output words in the Status-RAM.

Parameter description - inputs

IBS_IN

IBS_IN = Connection for the incoming remote bus part of INTERBUS
On the hardware, the male connector is on the top left of the module.
The module is connected here to the outgoing remote bus (IBS_OUT) of the master (1st module on the bus) or the preceding module (see diagram). The link can be via a line or via a variable. For the hardware, the type of connection corresponds to the INTERBUS cable between two bus devices.
Connection of two INTERBUS modules



PM_INx

PM_INx = Parameters for the input channels
x stands for the digit 1 to 4 which indicates the particular input channel.
These parameters are used to parameter the input channels.
The meaning of the parameter values can be found in the table below.
Table input range: Thermocouple

Parameter value	Cable break detection	Temperature unit	Input range
8705	no	1/10 degrees C	Thermocouple B
8961	yes		
8833	no	1/10 degrees F	
9089	yes		
4610	no	1/10 degrees C	Thermocouple E
4866	yes		
4738	no	1/10 degrees F	
4994	yes		
4611	no	1/10 degrees C	Thermocouple J
4867	yes		
4739	no	1/10 degrees F	
4995	yes		
4612	no	1/10 degrees C	Thermocouple K
4868	yes		
4740	no	1/10 degrees F	
4996	yes		
4613	no	1/10 degrees C	Thermocouple N
4869	yes		
4741	no	1/10 degrees F	
4997	yes		
8710	no	1/10 degrees C	Thermocouple R
8966	yes		
8838	no	1/10 degrees F	
9094	yes		
8711	no	1/10 degrees C	Thermocouple S
8967	yes		
8839	no	1/10 degrees F	
9095	yes		

Parameter value	Cable break detection	Temperature unit	Input range
8712	no	1/10 degrees C	Thermocouple T
8968	yes		
8840	no	1/10 degrees F	
9096	yes		

Table input range: IEC

Parameter value	Cable break detection	Temperature unit	Input range
2592	no	1/10 degrees C	IEC PT100 RTD 2 or 4 wire
2848	yes		
2720	no	1/10 degrees F	
2976	yes		
3616	no	1/10 degrees C	IEC PT100 RTD 3 wire
3872	yes		
3744	no	1/10 degrees F	
4000	yes		
545	no	1/10 degrees C	IEC PT100 RTD 2 or 4 wire
801	yes		
673	no	1/10 degrees F	
929	yes		
1569	no	1/10 degrees C	IEC PT100 RTD 3 wire
1825	yes		
1697	no	1/10 degrees F	
1953	yes		

Table input range: US/JIS

Parameter value	Cable break detection	Temperature unit	Input range
2656	no	1/10 degrees C	US/JIS PT100 RTD 2 or 4 wire
2912	yes		
2784	no	1/10 degrees F	
3040	yes		
3680	no	1/10 degrees C	US/JIS PT100 RTD 3 wire
3936	yes		
3808	no	1/10 degrees F	
4064	yes		
609	no	1/10 degrees C	US/JIS PT100 RTD 2 or 4 wire
865	yes		
737	no	1/10 degrees F	
993	yes		
1633	no	1/10 degrees C	US/JIS PT100 RTD 3 wire
1889	yes		
1761	no	1/10 degrees F	
2017	yes		

Table input range: DIN

Parameter value	Cable break detection	Temperature unit	Input range
2595	no	1/10 degrees C	DIN Ni100 RTD 2 or 4 wire
2851	yes		
2723	no	1/10 degrees F	
2979	yes		
3619	no	1/10 degrees C	DIN Ni100 RTD 3 wire
3875	yes		
3747	no	1/10 degrees F	
4003	yes		
546	no	1/10 degrees C	DIN Ni100 RTD 2 or 4 wire
802	yes		
674	no	1/10 degrees F	
930	yes		
1570	no	1/10 degrees C	DIN Ni100 RTD 3 wire
1826	yes		
1698	no	1/10 degrees F	
1954	yes		

Table input range: +/-25mV

Parameter value	Cable break detection	Temperature unit	Input range
8720	no		+/-25mV
8976	yes		
4625	no		+/-100mV
4881	yes		

Example:
Input 3 should be +/-25mV with wiring check.
PM_IN3 = "8976"

Parameter description - Outputs

IBS_OUT	IBS_OUT = Connection for the outgoing remote bus part of INTERBUS. On the hardware, the male connector is on the top right of the module. The module is connected to the incoming remote bus (IBS_IN) of the following module, either via a line or via a variable. For the hardware, the type of connection corresponds to the INTERBUS cable between two INTERBUS modules.
INx	INx = input channel x x stands for the digit 1 to 16 which indicates the particular input channel. The analog process values are read into the INTERBUS module via the relevant input (INx). **Note:** The values to be applied here are standardized, i.e. given as voltages in millivolts or as real values between -32000.0 and +32000.0. The input depends on the parametering of the particular channel.

ANA_4I_2O: Configuring the TIO-module BAM 096 00

19

Overview

Introduction

This chapter describes the block ANA_41_20.

What's in this chapter?

This chapter contains the following topics:

Brief description

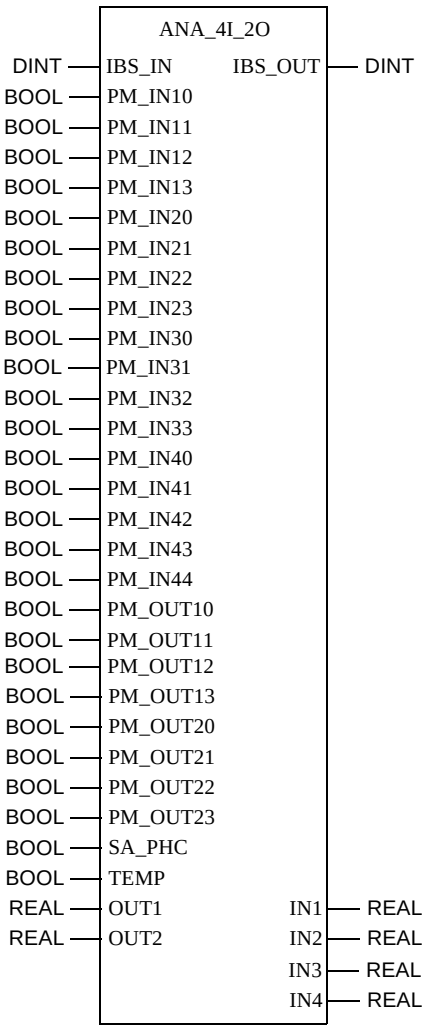
Function description

The ANA_4I_2O Function block is a software connection to the INTERBUS TIO/IS 170 BAM 096 00 hardware module.
The module has 4 analog inputs and 2 analog outputs. The function block must be configured corresponding to the hardware module (see *Detailed description, p. 99*).
A different designation was selected in order to achieve a clearer relation between the name and the function of the module.
EN and ENO can be configured as additional parameters.

Representation

Symbol

Block representation:



Parameter description

Block parameter description:

Parameter	Data type	Meaning
IBS_IN	DINT	Incoming INTERBUS
PM_IN10	BOOL	Input 1, parameter bit 0
:	:	:
PM_IN13	BOOL	Input 1, parameter bit 3
PM_IN20	BOOL	Input 2, parameter bit 0
:	:	:
PM_IN23	BOOL	Input 2, parameter bit 3
PM_IN30	BOOL	Input 3, parameter bit 0
:	:	:
PM_IN33	BOOL	Input 3, parameter bit 3
PM_IN40	BOOL	Input 4, parameter bit 0
:	:	:
PM_IN43	BOOL	Input 4, parameter bit 3
PM_OUT10	BOOL	Output 1, parameter bit 0
:	:	:
PM_OUT13	BOOL	Output 1, parameter bit 3
PM_OUT20	BOOL	Output 2, parameter bit 0
:	:	:
PM_OUT23	BOOL	Output 2, parameter bit 3
SA_PHC	BOOL	Compatibility of the analog value representation Schneider Automation (SA) = 1, Phoenix Contact (PHC) = 0
TEMP	BOOL	Type of temperature representation, 0 = C, 1 = F
OUT1	REAL	Output 1 of the TIO
OUT2	REAL	Output 2 of the TIO
IBS_OUT	DINT	Outgoing INTERBUS
IN1	REAL	Input 1 of the TIO
IN2	REAL	Input 2 of the TIO
IN3	REAL	Input 3 of the TIO
IN4	REAL	Input 4 of the TIO

Detailed description

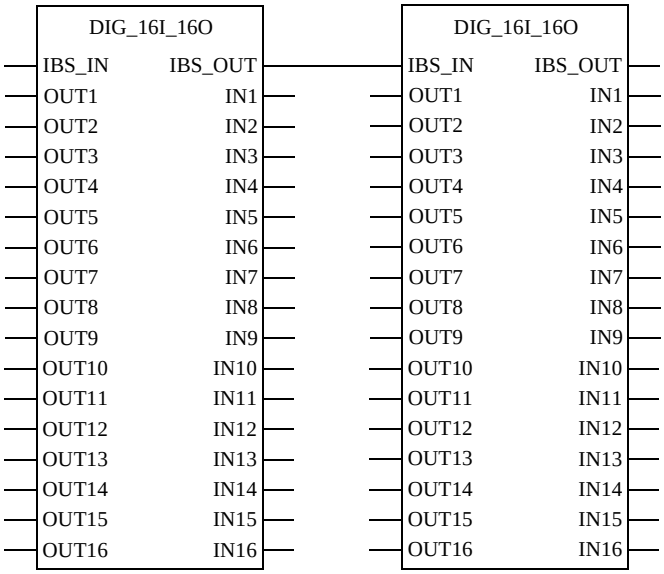
Detailed description

The block occupies 4 input words and 4 output words in the Status-RAM.

Parameter description - inputs

IBS_IN

IBS_IN = Connection for the incoming remote bus part of INTERBUS
On the hardware, the male connector is on the top left of the module.
The module is connected here to the outgoing remote bus (IBS_OUT) of the master (1st module on the bus) or the preceding module (see diagram). LinkConnection can be via a line or via a variable.line For the hardware, the type of connection corresponds to the INTERBUS cable between two bus devices.
Connection of two INTERBUS modules



PM_INxy

PM_INxy = Parameters for the input channels
x stands for the digit 1 to 4 which indicates the particular input channel.
y stands for the digit 0 to 3 which indicates the particular parameter bit.
Example: PM_IN23 = Parameters for input channel 2, bit 3
These parameters are used to parameter the input channels.
The meaning of the bits can be found in the table below.

Bit 3	Bit 2	Bit 1	Bit 0	Meaning
0	0	0	0	reserved; Channel inactive (default)
0	0	0	1	+/- 1 V
0	0	1	0	+/-20 mA (+/-5 V, when divided by 4)
0	0	1	1	+/- 10V
0	1	0	0	Channel inactive
0	1	0	1	0..0.1 V
0	1	1	0	0..0.5 V
0	1	1	1	0...10 V
1	0	0	0	reserved
1	0	0	1	0.2..0.1 V
1	0	1	0	420 mA (1...5 V, when divided by 4)
1	0	0	0	2...10 V
1	1	0	0	reserved
1	1	0	1	Pt100 with linearization
1	1	1	0	Ni100 with linearization
1	1	1	1	Resistance 0...2000 ohm

Example:
Input 3 should be 0 ...10 V.

- PM_IN30 = "1"
- PM_IN31 = "1"
- PM_IN32 = "1"
- PM_IN33 = "0"

Note: The reserved parameter codes are not accepted by the module, i.e. the last parameter used will still apply. The default parameters apply until a valid new parameter is entered.

PM_OUTxy

PM_OUTxy = Parameters for the output channels
x stands for the digit 1 or 2 which indicates the particular output channel.
y stands for the digit 0 to 3 which indicates the particular parameter bit.
Example: PM_OUT23 = Parameters for output channel 2, bit 3
These parameters are used to parameterize the output channels.
The meaning of the bits can be found in the table below.

Bit 3	**Bit 2**	**Bit 1**	**Bit 0**	**Meaning**	
				Output	**Output after bus interrupt (timeout)**
X	X	0	0	reserved; Channel inactive (default)	
0	0	0	1	0...20 mA	0 mA
0	0	1	0	4...20 mA	4 mA
0	0	1	1	+/-10 V/sensor supply	+0 V/sensor supply
0	1	0	0	0...20 mA	20 mA
0	1	1	1	4...20 mA	20 mA
0	1	1	0	+/-10 V/sensor supply	+10 V/sensor supply
1	0	0	1	0...20 mA	freezes
1	0	1	0	4...20 mA	freezes
1	0	1	1	+/-10 V/sensor supply	freezes
1	1	X	X	reserved	

Example:
Output 1 should be 0 20mA and be set to 0mA for bus failure.
- PM_OUT10 = "1"
- PM_OUT11 = "0"
- PM_OUT12 = "0"
- PM_OUT13 = "0"

Note: The reserved parameter codes are not accepted by the module, i.e. the last parameter used will still apply. The default parameters apply until a valid new parameter is entered.

PM_OUTxy

SA_PHC = Compatibility of the analog value representation
This parameter is used to enter the compatibility of the analog value representation; a "1" stands for Schneider Automation compatibility and a "0" for Phoenix Contact. The setting applies for inputs and outputs.

TEMP

TEMP = Type of temperature representation
This parameter can be used to select the temperature representation. If the bit is set to "0", the temperature is given in degrees Celsius, if it is "1", it is in degrees Fahrenheit.

OUTx

OUTx = output channel x
x stands for the number 1 or 2 which refers to the corresponding output. The analog values to be produced via the INTERBUS module are supplied to the process via the relevant output (OUTx).

Note: The values to be applied here are standardized, i.e. given as voltage in volts or as current in milliamperes. The input as current or voltage depends on the parametering of the particular channel.

Parameter description - Outputs

IBS_OUT

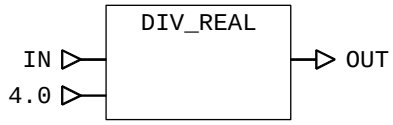
IBS_OUT = Connection for the outgoing remote bus part of INTERBUS
On the hardware, the male connector is on the top right of the module.
The module is connected to the incoming remote bus (IBS_IN) of the following module, either via a line or via a variable. For the hardware, the type of connection corresponds to the INTERBUS cable between two INTERBUS modules.

INx

INx = input channel x
x stands for the number between 1 and 4 designating the corresponding input channel. The analog process values of the INTERBUS module are read via the corresponding input (INx).

Note: The values to be applied here are standardized, i.e. given as voltage in volts or as current in milliamperes. The input as current or voltage depends on the parameters set for the particular channel. .

If a channel is parametered in the +/-5 V- or 1...5 V range, the incoming values are given in milliamperes. To obtain these values as voltage, divide by 4.0 (see diagram).
Scaling an analog value



IN Analog value in mA
OUT Analog value in V

ANA_4I_2O_C: Configuring the TIO-module BAM 096 00

20

Overview

Introduction

This chapter describes the block ANA_41_20_C.

What's in this chapter?

This chapter contains the following topics:

Brief description

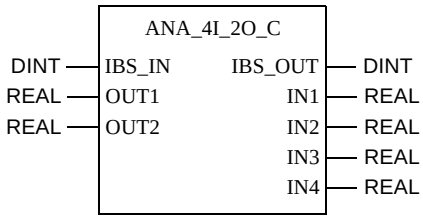
Function description

The function block ANA_4I_2O_C is a software connection to the INTERBUS TIO/IS 170 BAM 096 00 hardware module with preset current.
The module has 4 analog inputs and 2 analog outputs.
EN and ENO can be configured as additional parameters.

Representation

Symbol

Block representation:



Parameter description

Block parameter description:

Parameter	Data type	Meaning
IBS_IN	DINT	Incoming INTERBUS
OUT1	REAL	Output 1 of the TIO Range: 0...20 mA
OUT2	REAL	Output 2 of the TIO Range: 0...20 mA
IBS_OUT	DINT	Outgoing INTERBUS
IN1	REAL	Input 1 of the TIO range: +/-20 mA
IN2	REAL	Input 2 of the TIO range: +/-20 mA
IN3	REAL	Input 3 of the TIO range: +/-20 mA
IN4	REAL	Input 4 of the TIO range: +/-20 mA

Detailed description

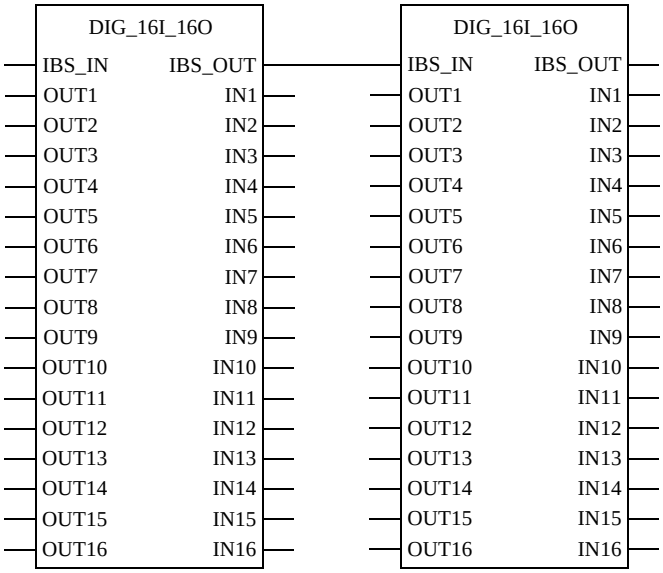
Detailed description

The function block ANA_4I_2O_C is a special version of ANA_4I_2O. The block is already set for current response (C = current), so the user does not need to set parameters for this.
On this block, all input channels are set to +/- 20 mA; the output channels are set to 0...20mA, Timeout A: set to 0 mA.
The block occupies 4 input words and 4 output words in the Status-RAM.

Parameter description - inputs

IBS_IN

IBS_IN = Connection for the incoming remote bus part of INTERBUS
On the hardware, the male connector is on the top left of the module.
The module is connected here to the outgoing remote bus (IBS_OUT) of the master (1st module on the bus) or the preceding module (see diagram). The link can be made via a line or via a variable. For the hardware, the type of connection corresponds to the INTERBUS cable between two bus devices.
Connection of two INTERBUS modules



OUTx

OUTx = output channel x
x stands for the number 1 or 2 which refers to the corresponding output. The analog values to be produced via the INTERBUS module are supplied to the process via the relevant output (OUTx).

Note: The values to be applied here are standardized, i.e. given as current in milliamperes.

Parameter description - Outputs

IBS_OUT

IBS_OUT = Connection for the outgoing remote bus part of INTERBUS
On the hardware, the male connector is on the top right of the module.
The module is connected to the incoming remote bus (IBS_IN) of the following module, either via a line or via a variable. For the hardware, the type of connection corresponds to the INTERBUS cable between two INTERBUS modules.

INx

INx = input channel x
x stands for the number between 1 and 4 designating the corresponding input channel. The analog process values of the INTERBUS module are read via the corresponding input (INx).

Note: The incoming values are standardized, i.e. given as current in milliamperes.

ANA_4I_2O_V: Configuring the TIO-module BAM 096 00

21

Overview

Introduction

This chapter describes the block ANA_41_20_V.

What's in this chapter?

This chapter contains the following topics:

Brief description

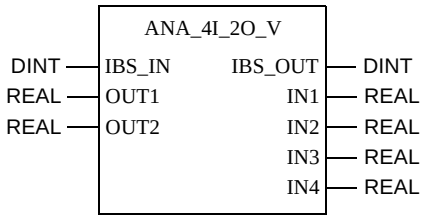
Function description

The function block ANA_4I_2O_V is a software connection to the INTERBUS TIO/IS 170 BAM 096 00 hardware module with preset voltage.
The module has 4 analog inputs and 2 analog outputs.
EN and ENO can be configured as additional parameters.

Representation

Symbol

Block representation:



Parameter description

Block parameter description:

Parameter	Data type	Meaning
IBS_IN	DINT	Incoming INTERBUS
OUT1	REAL	Output 1 of the TIO Range: +/-10 V
OUT2	REAL	Output 2 of the TIO Range: +/-10 V
IBS_OUT	DINT	Outgoing INTERBUS
INT1	REAL	Input 1 of the TIO Range: +/-10 V
INT2	REAL	Input 2 of the TIO Range: +/-10 V
INT3	REAL	Input 3 of the TIO Range: +/-10 V
INT4	REAL	Input 4 of the TIO Range: +/-10 V

Detailed description

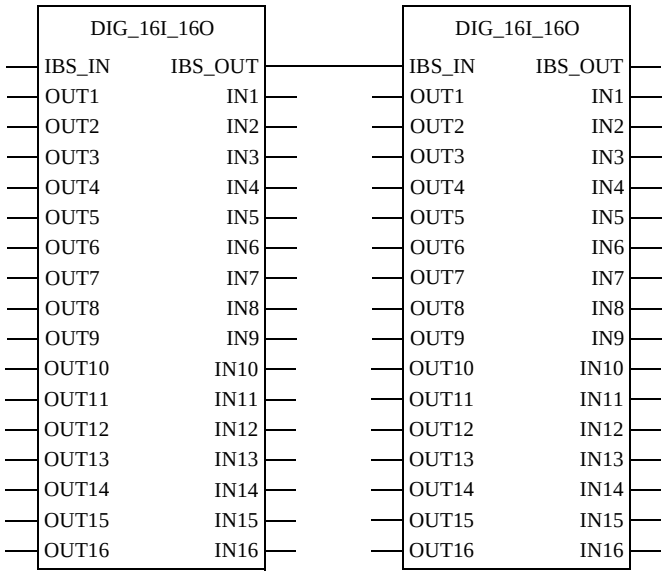
Detailed description

The function block ANA_4I_2O_V is a special version of ANA_4I_2O.
The block is already set for voltage response (V = voltage), so the user does not need to set parameters for this. With these blocks, all input channels and output channels are set to +/- 10 V and the Timeout A to 0 V.
The block occupies 4 input words and 4 output words in the Status-RAM.

Parameter description - inputs

IBS_IN

IBS_IN = Connection for the incoming remote bus part of INTERBUS
On the hardware, the male connector is on the top left of the module.
The module is connected here to the outgoing remote bus (IBS_OUT) of the master (1st module on the bus) or the preceding module (see diagram). LinkConnection can be via a line or via a variable.line For the hardware, the type of connection corresponds to the INTERBUS cable between two bus devices.
Connection of two INTERBUS modules



OUTx

OUTx = output channel x
x stands for the digit 1 or 2 which indicates the particular output channel.
The analog values to be produced via the INTERBUS module are supplied to the process via the relevant output (OUTx).

Note: The values to be applied here are standardized, ie. given as voltage in V.

Parameter description - Outputs

IBS_OUT

IBS_OUT = Connection for the outgoing remote bus part of INTERBUS
On the hardware, the male connector is on the top right of the module.
The module is connected to the incoming remote bus (IBS_IN) of the following module, either via a line or via a variable. For the hardware, the type of connection corresponds to the INTERBUS cable between two INTERBUS modules.

INx

INx = input channel x
x stands for the digit 1 to 4 which indicates the particular input channel
The analog process values are read into the INTERBUS module via the relevant input (INx).

Note: The incoming values are standardized, i.e. given as voltage in V.

ANA_4O: Configuring the module BAO 126 00

22

Overview

Introduction

This chapter describes the block ANA_40.

What's in this chapter?

This chapter contains the following topics:

Brief description

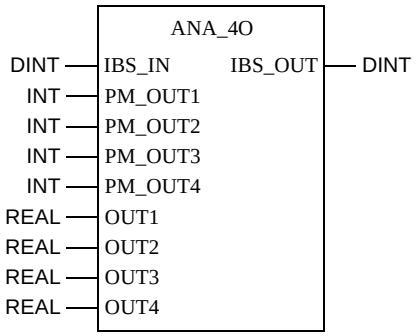
Function description

The ANA_40 Function block is a software connection to the INTERBUS TIO/IS 170 BAO 126 00 hardware module.
The function block has 4 analog outputs. The function block must be parametered in the same way as the hardware module.
EN and ENO can be projected as additional parameters.

Representation

Symbol

Block representation:



Parameter description

Block parameter description:

Parameter	Data type	Meaning
IBS_IN	DINT	Incoming INTERBUS
PM_OUT1	INT	Parameter output 1
:	:	:
PM_OUT4	INT	Parameter output 4
OUT1	REAL	Output 1 of the module
:	:	:
OUT4	REAL	Output 4 of the module
IBS_OUT	DINT	Outgoing INTERBUS

Detailed description

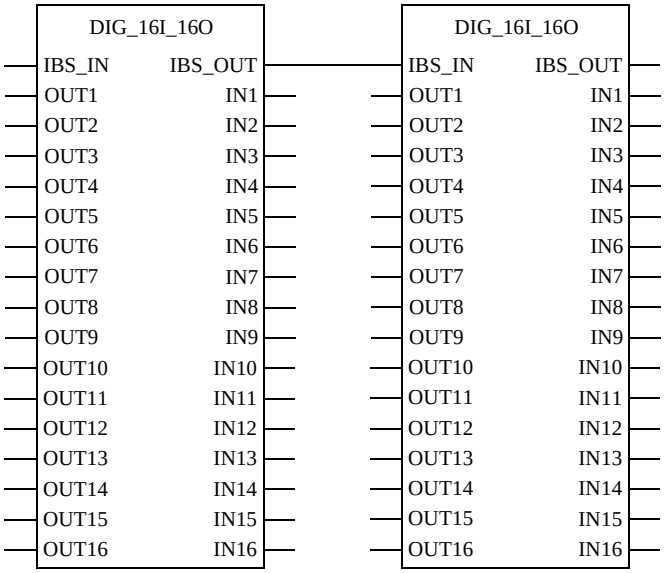
Detailed description

The function block occupies 5 output words in the Status-RAM.

Parameter description - inputs

IBS_IN

IBS_IN = Connection for the incoming remote bus part of INTERBUS
On the hardware, the male connector is on the top left of the module.
The module is connected here to the outgoing remote bus (IBS_OUT) of the master (1st module on the bus) or the preceding module (see diagram). The link can be made via a line or via a variable. For the hardware, the type of connection corresponds to the INTERBUS cable between two bus devices.
Connection of two INTERBUS modules



PM_OUTx

PM_OUTx = Parameters for the output channels
x stands for the digit 1 or 4 which indicates the particular output channel.

Example:
PM_OUT2 = Parameters for output channel 2

These parameters are used to set parameters for the output channels.
The meaning of the parameter values can be found in the table below.

Parameter value	Meaning
0	reserved; Channel inactive (default)
1	0...20mA; Timeout A: 0mA
2	4...20mA; Timeout A: 4mA
3	+/- 10V; Timeout A: 0V
5	0...20mA; Timeout A: 20mA
6	4...20mA; Timeout A: 20mA
7	+/- 10V; Timeout A: +10V
9	0...20mA; Timeout A: freezes
10	4...20mA; Timeout A: freezes
11	+/- 10V; Timeout A: freezes

A = Output after bus interrupt

Note: All other parameter values are reserved.

Example:
Output 1 should be 0 ...20mA and set to 0mA for bus failure.
PM_OUT1 = "1"

Note: The reserved parameter codes are not accepted by the module, i.e. the last parameter used will still apply. The default parameters apply until a valid new parameter is entered.

OUTx

OUTx = output channel x
x stands for the digit 1 or 4 which indicates the particular output channel.
The analog values to be produced via the INTERBUS module are supplied to the process via the relevant output (OUTx).

Note: The values to be applied here are standardized, i.e. given as voltage in volts or as current in milliamperes. The input as current or voltage depends on the parametering of the particular channel.

Parameter description - Outputs

IBS_OUT

IBS_OUT = Connection for the outgoing remote bus part of INTERBUS
On the hardware, the male connector is on the top right of the module.
The module is connected to the incoming remote bus (IBS_IN) of the following module, either via a line or via a variable. For the hardware, the type of connection corresponds to the INTERBUS cable between two INTERBUS modules.

ANA_8I: Configuring the module AAI 030 00, BAI 036 00

23

Overview

Introduction

This chapter describes the block ANA_8I.

What's in this chapter?

This chapter contains the following topics:

Brief description

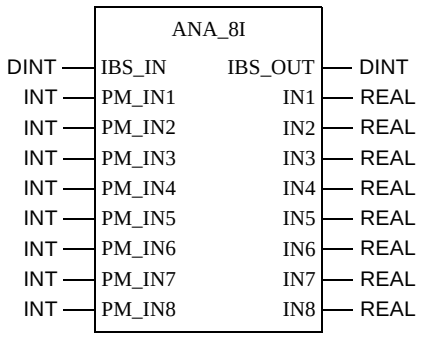
Function description

The ANA_81 Function block is a software connection to the INTERBUS, TIO/IS 170 BAI 036 00 and Momentum/IS 170 AAI 030 00 hardware modules.
The module has 8 analog inputs. The function block must be configured in the same way as the hardware module.
EN and ENO can be configured as additional parameters.

Representation

Symbol

Block representation:



Parameter description

Block parameter description:

Parameter	Data type	Meaning
IBS_IN	DINT	Incoming INTERBUS
PM_IN1	INT	Parameter for input 1
:	:	:
PM_IN8	INT	Parameter for input 8
IBS_OUT	DINT	Outgoing INTERBUS
IN1	REAL	Input 1 of the module
:	:	:
IN8	REAL	Input 8 of the module

Detailed description

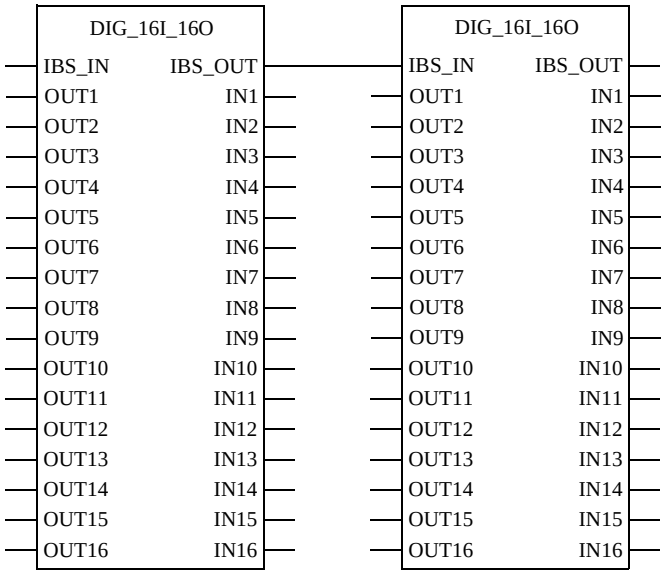
Detailed description

The function block occupies 8 input words and 8 output words in the Status-RAM.

Parameter description - inputs

IBS_IN

IBS_IN = Connection for the incoming remote bus part of INTERBUS
On the hardware, the male connector is on the top left of the module.
The module is connected here to the outgoing remote bus (IBS_OUT) of the master (1st module on the bus) or the preceding module (see diagram). The link can be made via a line or via a variable. For the hardware, the type of connection corresponds to the INTERBUS cable between two bus devices.
Connection of two INTERBUS modules



PM_INx

PM_INx = Parameters for the input channels
x stands for the digit 1 to 8 which indicates the particular input channel.
These parameters are used to configure the input channels.
The meaning of the parameter values can be found in the table below.

Parameter value	Meaning
0	Channel inactive (default)
2	+/- 20mA (+/- 5 V, when divided by 4)
3	+/- 10V
4	Channel inactive
6	020mA (0...5 V, when divided by 4)
7	0...10V
10	420mA (1...5 V, when divided by 4)

A = Output after bus interrupt

Note: All other parameter values are reserved.

Example:
Input 3 should 4 ...20mA.
PM_IN3 = "10"

Note: The reserved parameter codes are not accepted by the module, i.e. the last parameter used will still apply. The default parameters apply until a valid new parameter is entered.

Parameter description - Outputs

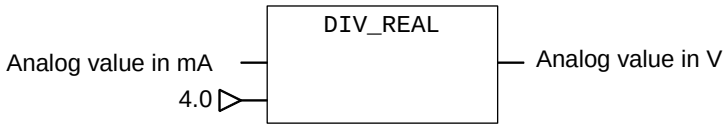
IBS_OUT

IBS_OUT = Connection for the outgoing remote bus part of INTERBUS
On the hardware, the male connector is on the top right of the module.
The module is connected to the incoming remote bus (IBS_IN) of the following module, either via a line or via a variable. For the hardware, the type of connection corresponds to the INTERBUS cable between two INTERBUS modules.

INx

INx = input channel x
x stands for the digit 1 to 8 which indicates the particular input channel.
The analog process values are read into the INTERBUS module via the relevant input (INx).

Note: The values to be applied here are standardized, i.e. given as voltage in volts or as current in milliamperes. The input as current or voltage depends on the configuration of the particular channel. If a channel is configured in the +/-5V- or 1...5V range, the incoming values are given in milliamperes. To obtain these values as voltage, divide by 4.0.
Scaling an analog value



ARI030: Configuring the Quantum module ARI 030 10

24

Overview

Introduction

This chapter describes the block ARI030.

What's in this chapter?

This chapter contains the following topics:

Brief description

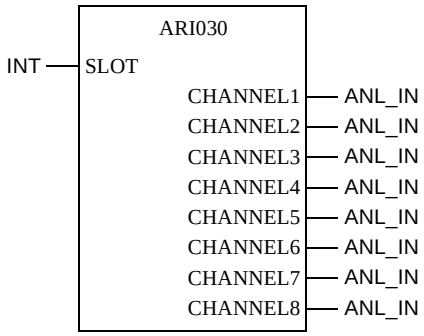
Function description

The Function block is used to edit the configuration data of an ARI 030 10 Quantum module for subsequent use by the scaling EFBs.
This module has 8 resistor temperature sensor input channels (RTD) for the processing of four-wire RTD sensors.
For the configuration of an ARI 030 10 the function block in the configuration section is connected to the corresponding SLOT output of the QUANTUM or DROP Function block. The 3x references specified in the I/O map are automatically assigned internally to the individual channels and can therefore only be occupied by Unlocated variables.
The analog values can be further processed in Scaling Sections using the I_DEBUG, I_NORM, I_NORM_WARN, I_PHYS, I_PHYS_WARN and I_RAW Function blocks.
EN and ENO can be projected as additional parameters.

Representation

Symbol

Block representation:



Parameter description

Block parameter description:

Parameter	Data type	Meaning
SLOT	INT	Slot
CHANNEL1	ANL_IN	Channel 1
CHANNEL2	ANL_IN	Channel 2
CHANNEL3	ANL_IN	Channel 3
CHANNEL4	ANL_IN	Channel 4
CHANNEL5	ANL_IN	Channel 5
CHANNEL6	ANL_IN	Channel 6
CHANNEL7	ANL_IN	Channel 7
CHANNEL8	ANL_IN	Channel 8

Runtime error

Runtime error

If no ARI 030 10 module has been configured for the specified SLOT input, an error message appears.
The range warning for the channels can be evaluated through the I_NORM_WARN or I_PHYS_WARN Function block.
The status message "Open circuit or range violation on channel" can be requested via the status register defined in the I/O map.

ATI030: Configuring the Quantum module ATI 030 00

25

Overview

Introduction

This chapter describes the block ATI030.

What's in this chapter?

This chapter contains the following topics:

Brief description

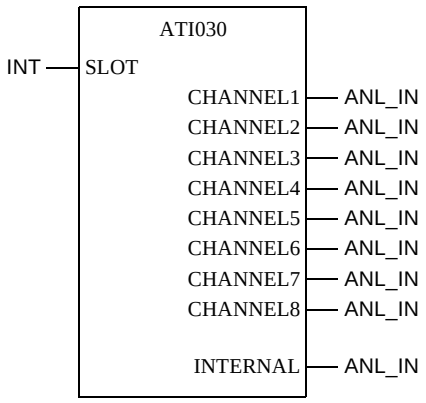
Function description

The Function block is used to edit the configuration data of an ATI 030 00 Quantum module for subsequent use by the scaling EFBs.
This module has 8 thermocouple input channels.
For the configuration of an ATI 030 00 the function block in the configuration section is connected to the corresponding SLOT output of the QUANTUM or DROP Function block. The 3x references specified in the I/O map are automatically assigned internally to the individual channels and can therefore only be occupied by Unlocated variables.
The analog values can be further processed in Scaling sections using the fuction blocks I_DEBUG, I_NORM, I_NORM_WARN and I_RAW.
EN and ENO can be configured as additional parameters.

Representation

Symbol

Block representation:



Parameter description

Block parameter description:

Parameter	Data type	Meaning
SLOT	INT	Slot
CHANNEL1	ANL_IN	Channel 1
CHANNEL2	ANL_IN	Channel 2
CHANNEL3	ANL_IN	Channel 3
CHANNEL4	ANL_IN	Channel 4
CHANNEL5	ANL_IN	Channel 5
CHANNEL6	ANL_IN	Channel 6
CHANNEL7	ANL_IN	Channel 7
CHANNEL8	ANL_IN	Channel 8
INTERNAL	ANL_IN	Temperature of module

Runtime error

Runtime error

If no ATI 030 00 module has been configured for the specified SLOT input, an error message appears.
The rangewarning for the input channels can be evaluated using the function block I_NORM_WARN or I_PHYS_WARN.
The status information "Range violation on channel" can be collected via the status register defined in the I/O map.

AVI030: Configuring the Quantum module AVI 030 00

26

Overview

Introduction

This chapter describes the block AVI030.

What's in this chapter?

This chapter contains the following topics:

Brief description

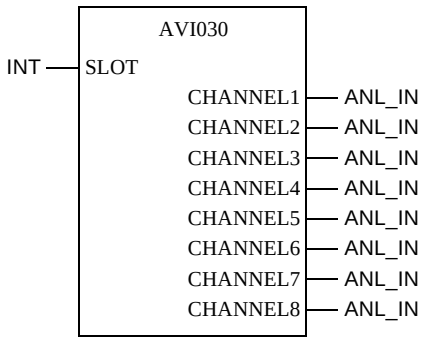
Function description

The Function block is used to edit the configuration data of an AVI 030 00 Quantum module for subsequent use by the scaling EFBs.
This module has 8 bipolar input channels for mixed voltage and current processing.
For the configuration of an AVI 030 00 the function block in the configuration section is connected to the corresponding SLOT output of the QUANTUM or DROP Function block. The 3x references specified in the I/O map are automatically assigned internally to the individual channels and can therefore only be occupied by Unlocated variables.
The analog values can be further processed inScaling sections using the I_DEBUG, I_NORM, I_NORM_WARN, I_PHYS, I_PHYS_WARN, I_RAW, I_SCALE and I_SCALE_WARN Function blocks.
EN and ENO can be projected as additional parameters.

Representation

Symbol

Block representation:



Parameter description

Block parameter description:

Parameter	Data type	Meaning
SLOT	INT	Slot
CHANNEL1	ANL_IN	Channel 1
CHANNEL2	ANL_IN	Channel 2
CHANNEL3	ANL_IN	Channel 3
CHANNEL4	ANL_IN	Channel 4
CHANNEL5	ANL_IN	Channel 5
CHANNEL6	ANL_IN	Channel 6
CHANNEL7	ANL_IN	Channel 7
CHANNEL8	ANL_IN	Channel 8

Runtime error

Runtime error

If no AVI 030 00 module has been configured for the specified SLOT input, an error message appears.
The range warning for the channels can be evaluated using the I_NORM_WARN, I_PHYS_WARN or I_SCALE_WARN Function blocks.
The status information "Open circuit or range violation on channel" can be collected via the status register defined in the I/O map.

AVO020: Configuring the Quantum module AVO 020 00

27

Overview

Introduction

This chapter describes the block AVO20.

What's in this chapter?

This chapter contains the following topics:

Brief description

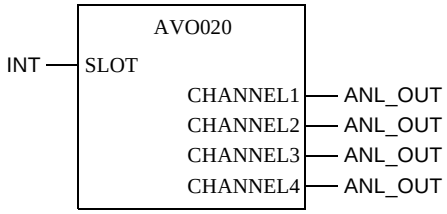
Function description

The Function block is used to edit the configuration data of the AVO 020 00 Quantum module for subsequent use by the scaling EFBs.
This module has 4 voltage output channels with mixed modes and levels.
For the configuration of an AVO 020 the function block in the configuration section is connected to the corresponding SLOT output of the QUANTUM or DROP Function block. The 4x references specified in the I/O map are automatically assigned internally to the individual channels and can therefore only be occupied by Unlocated variables.
The analog values can be further processed in Scaling Sections using the O_DEBUG, O_NORM, O_RAW and O_SCALE Function blocks.
EN and ENO can be projected as additional parameters.

Representation

Symbol

Block representation:



Parameter description

Block parameter description:

Parameter	Data type	Meaning
SLOT	INT	Slot
CHANNEL1	ANL_OUT	Channel 1
CHANNEL2	ANL_OUT	Channel 2
CHANNEL3	ANL_OUT	Channel 3
CHANNEL4	ANL_OUT	Channel 4

Runtime error

Runtime error

If no AVO 020 00 module has been configured for the specified SLOT input, an error message appears.

BKF_201: Configuring the Compact module BKF 201

28

Overview

Introduction

This chapter describes the block BKF_201.

What's in this chapter?

This chapter contains the following topics:

Brief description

Function description

The BKF_201 Function block is the software connection to an INTERBUS BKF 201 master module.
It ensures that the data on the INTERBUS is correctly transferred to and read by the relevant module. The BKF 201 controls the bus and monitors operational performance.

Varying quantities of INTERBUS data can be transferred/read according to the slot.

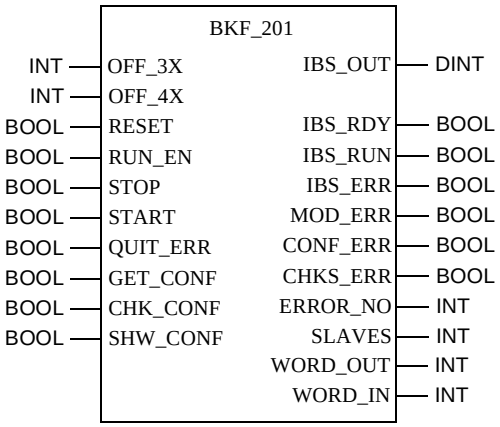
- Slot in main rack (BKF201(64 W))
  63 input words and 63 output words
- Slot in rack (BKF201(16 W))
  15 input words and 15 output words

The BKF 201 occupies 16 or 64 input words and 16 or 64 output words in PLC memory.
The first input word and the first output word are occupied by the BKF 201 itself. These contain the BKF 201 control bits and status bits. The remaining 15 or 63 input words and 15 or 63 output words contain the I/O data for the INTERBUS modules. EN and ENO can be projected as additional parameters.

Representation

Symbol

Block representation:



Parameter description

Block parameter description:

Parameter	Data type	Meaning
OFF_3X	INT	Offset for 3x address
OFF_4X	INT	Offset for 4x address
RESET	BOOL	Reset and reconfigure the BKF 201
RUN_EN	BOOL	CPU STOP routine (acc. to operations software Version 1.01)
STOP	BOOL	Emergency stop
START	BOOL	Start cycle
QUIT_ERR	BOOL	Error acknowledgment
GET_CONF	BOOL	Get configuration
CHK_CONF	BOOL	Check configuration
SHW_CONF	BOOL	Show configuration
IBS_OUT	DINT	Outgoing INTERBUS
IBS_RDY	BOOL	INTERBUS ready
IBS_RUN	BOOL	INTERBUS data is being transmitted
IBS_ERR	BOOL	INTERBUS faulty
MOD_ERR	BOOL	Module error
CONF_ERR	BOOL	INTERBUS configuration faulty
CHKS_ERR	BOOL	Checksum error
ERROR_NO	INT	Error number
SLAVES	INT	Number of INTERBUS devices
WORD_OUT	INT	Number of process data output words
WORD_IN	INT	Number of process data input words

Detailed description

Detailed description

The BKF_201 Function block works in the same way as its hardware counterpart. However, it has been possible to greatly simplify operation through the software connection. The module only processes words. It occupies either 64 (main rack) or 16 (rack) 3x addresses and either 64 (main rack) or 16 (rack) 4x addresses in the PLC.

Note: The dip switches on the back of the BKF 201 can be used to select the data width (16/64 words) of the module (see HW description for the module).

Parameter description - inputs

The BKF_201 inputs in Concept differ only slightly from those for the hardware module. Compared to the hardware, inputs OFF_3X, OFF_4X and RESET are new. The input abbreviations are assigned the following module functions or control bits.

OFF_3X, OFF_4X

OFF_3X = Offset 3x address
OFF_4X = Offset 4x address
On the function block, the relevant address offsets for 3x and 4x addresses are given at the two inputs (start addresses of the HW modules in I/O map).

Example:
The BFK 201 (16 W) is entered in the PLC configurator, as shown in the table.

Slot	Module	In Ref	In End	Out Ref	Out End
4	BKF201(16 W)	300020	300035	400020	400035

If the initial addresses for the BKF 201 are 3:20 for the input words and 4:20 for the output words, then
- OFF_3X = 20 and
- OFF_4X = 20.

RESET

RESET = reset and reconfigure the BKF 201
If RESET is set to "1", the BKF 201 is reset and a new INTERBUS configuration is started. RESET also performs error acknowledgment, see also "Time diagram for error acknowledgment".

RUN_EN

RUN_EN = CPU STOP routine (according to operations software Version 1.01)
0 = the IOBUS is deactivated if the CPU is stopped
1 = the IOBUS remains active if the CPU is stopped

STOP

STOP = emergency stop
If this bit is set, the INTERBUS stops immediately and all outputs are set to zero. When this bit is set, the other control bits have no effect.

START

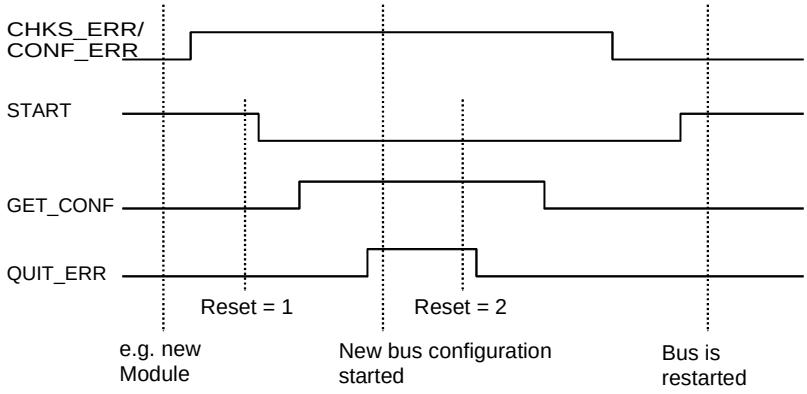
START = start cycle
The master transmits data to the INTERBUS devices when this bit is set. If the bit is deleted, the device outputs remain at their last value. Input information is maintained.

QUIT_ERR

QUIT_ERR = error acknowledgment
As its name suggests, this is used to acknowledge errors.
This bit should not be set permanently, otherwise any errors occurring will be immediately acknowledged and therefore deleted.

GET_CONF

GET_CONF = get configuration
When this bit is set, a new INTERBUS configuration is started. This is useful when the bus structure has been changed and the master has set either the CONF_ERR or CHKS_ERR status bit (see Time diagram). All other bits must first be set to zero. After the GET_CONF has been set, the QUIT_ERR should be set in a way that ensures both control bits are present simultaneously.
Don't forget to reset both of bits again and to restart the bus.
Time diagram



CHK_CONF

CHK_CONF = check configuration
If this bit is set, the BKF 201 compares the actual configuration checksum with the desired configuration checksum. If they differ, the bus is stopped and CONF_ERR is set.

SHW_CONF	SHW_CONF = Show configuration The actual INTERBUS configuration is displayed; all other control bits must be "0". The configuration is displayed at the outputs SLAVES, WORD_OUT and WORD_IN. Until this function is used, the outputs are set to "0".
Parameter description - Outputs	The BKF_201 outputs in Concept differ only slightly from those for the hardware module. Compared to the hardware, the status indicators CHKS_ERR and ERROR_NO are new or have been changed. Outputs IBS_RDY to CONF_ERR correspond to the status bits of the BFK 201. The input abbreviations are assigned the following module functions or control bits.
IBS_OUT	IBS_OUT = Connection for the outgoing remote bus part of INTERBUS INTERBUS connection on the front panel of the BKF 201. From here, the first module on INTERBUS is connected to the master, either via a line or via a variable. For the hardware, the type of connection corresponds to the INTERBUS cable from the master to the first module on INTERBUS.
IBS_RDY	IBS_RDY = INTERBUS ready INTERBUS is ready and error-free. This bit corresponds to LED No. 3 on the BKF 201.
IBS_RUN	IBS_RUN = INTERBUS data is being transmitted INTERBUS is operating without error, process data is being exchanged. This bit corresponds to LED No. 4 on the BKF 201.
IBS_ERR	IBS_ERR = INTERBUS error Indicates a bus error. This can be due to an open circuit, short circuit or voltage failure on a device or break-down during data transmission. This bit corresponds to LED No. 5 on the BKF 201.
MOD_ERR	MOD_ERR = module error An error has occurred on an INTERBUS module. The error does not stop INTERBUS. This bit corresponds to LED No. 6 on the BKF 201.
CONF_ERR	CONF_ERR = configuration of INTERBUS faulty This error can be due to wiring errors, changes to the configuration during operation, devices not ready or similar. This bit corresponds to LED No. 7 on the BKF 201.

CHKS_ERR

CHKS_ERR = checksum error
The INTERBUS configuration was changed during loss of voltage on the BKF 201. For an error to be detected, data loss in the PLC must be excluded. The checksum error does not occur when a program is loaded for the first time. The checksum error is not detected by the BKF 201 hardware, which can in certain circumstances lead to errors on INTERBUS, because the BKF 201 configures itself after voltage recovery. Error detection is possible because the old checksum is stored and compared with the new checksum. The error is deleted using QUIT_ERR or RESET.

ERROR_NO

ERROR_NO = Device error number
Indicates the device number of a faulty device on the bus. This indication corresponds to LEDs numbered. 14 (Significance 1) to 21 (Significance 80) on the BKF 201.
Example:

- Bus connection between device 1 and device 2 interrupted.
  Display: 2
- Voltage failure on device 1.
  Display: 1

SLAVES

SLAVES = number of INTERBUS devices
Indicates the number of bus devices connected to the master.

WORD_OUT

WORD_OUT = number of process data output words
Indicates the number of output words (4x register) occupied in the master. If the value indicated is 63 (main rack), or 15 (rack) then no additional INTERBUS module which occupies one or more output word in the master can be connected to INTERBUS.

WORD_IN

WORD_IN = number of process data input words
Indicates the number of input words (3x register) occupied in the master. If the value indicated is 63 (main rack), or 15 (rack) then no additional INTERBUS module which occupies more than one input word in the master can be connected to INTERBUS.

Runtime error

Runtime error

An error message (E_INPUT_VALUE_OUT_OF_RANGE) appears if the offset for the 3x or 4x addresses < 0 and/or more than the maximum permissible value.

BNO_671: Configuring the TIO-module BNO 671 00

29

Overview

Introduction

This chapter describes the block BNO_671.

What's in this chapter?

This chapter contains the following topics:

Brief description

Function description

The BNO_671 Function block in Concept functions in the same way as its hardware counterpart.
By programming the bus as a function block in Concept, it can be divided into different segments even without 170 BNO 671 00 hardware. As only unidirectional connections are possible in Concept, unlike with hardware, the IBS_OUT output of the last module in the remote bus spur must be connected to the LRB_IN input of the BNO_671. The module does not occupy a word in the master.

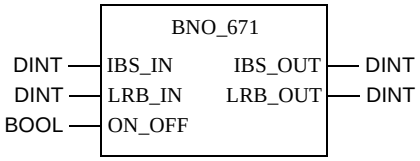
Note: If the 170 BNO 671 00 is used, its status cannot be indicated using the BNO_671.

EN and ENO can be projected as additional parameters.

Representation

Symbol

Block representation:



Parameter description

Block parameter description:

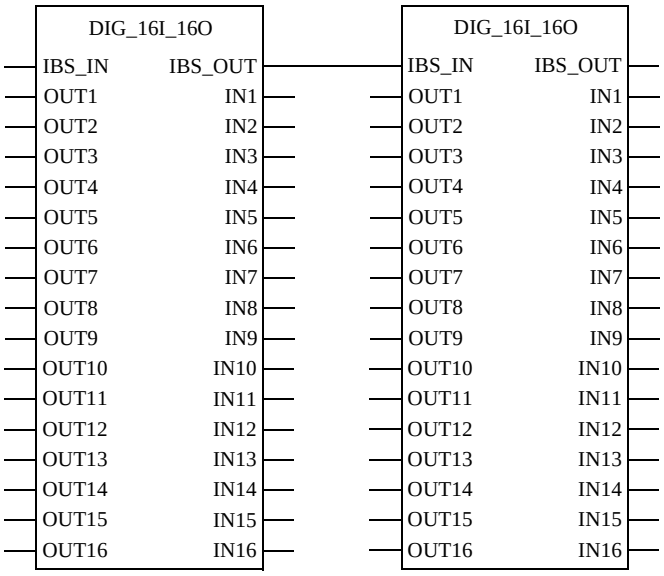
Parameters	Data type	Meaning
IBS_IN	DINT	Incoming INTERBUS
LRB_IN	DINT	Incoming remote bus branch and/or local bus (Local and/or Remote Bus)
ON_OFF	BOOL	On or Off switch for the remote bus branch output
IBS_OUT	DINT	Outgoing INTERBUS
LRB_OUT	DINT	Outgoing remote bus branch and/or local bus (Local and/or Remote Bus)

Detailed description

Parameter description - inputs

IBS_IN

IBS_IN = Connection for the incoming remote bus part of INTERBUS
On the hardware, the male connector is on the top left of the module.
The module is connected here to the outgoing remote bus (IBS_OUT) of the master (1st module on the bus) or the preceding module (see diagram). The link can be made via a line or via a variable. For the hardware, the type of connection corresponds to the INTERBUS cable between two bus nodes.
Connection of two INTERBUS modules



LRB_IN

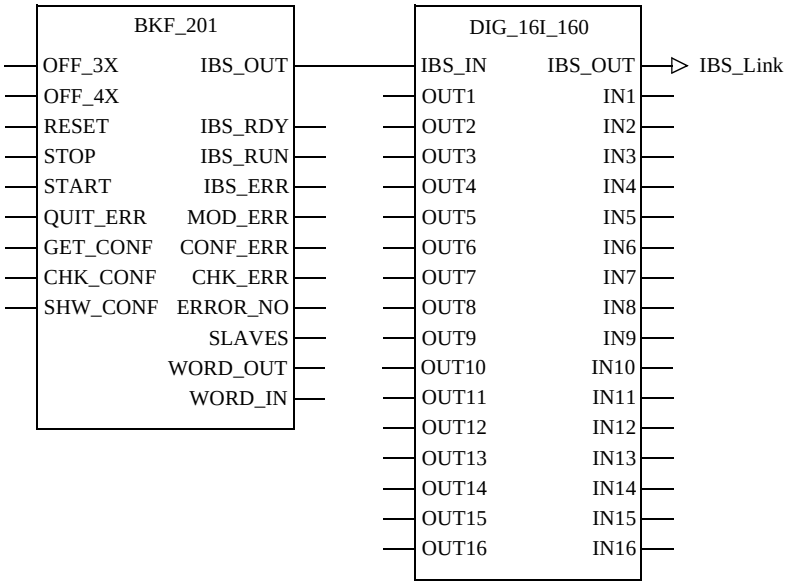
LRB_IN = Local remote bus input (incoming remote bus branch)
The IBS_OUT output of the last module on the remote bus branch is connected to this input to ensure that data is passed on correctly in INTERBUS.

ON_OFF	ON_OFF = Input for switching the remote bus branch on and off A"0" at input ON_OFF means that all outputs of an INTERBUS module are set to "0" on the remote bus branch. A "1" means that the values predefined in the program (e.g. UNI_I_O) are written to the corresponding 4x register and transmitted to the process. ON_OFF has no effect on the inputs of the INTERBUS module. Data traffic in the remote bus branch is independent from the valence of ON_OFF, i.e. it is not affected.
Parameter description - Outputs	
IBS_OUT	IBS_OUT = Connection for the outgoing remote bus part of INTERBUS On the hardware, the male connector is on the top right of the module. The module is connected to the incoming remote bus (IBS_IN) of the following module, either via a line or via a variable. For the hardware, the type of connection corresponds to the INTERBUS cable between two INTERBUS modules.
LRB_OUT	LRB_OUT = Connection for the outgoing INTERBUS remote bus branch On the hardware, the male connector is on the front panel of the module and is marked as Local Remote Bus. The module is connected to the incoming remote bus (IBS_IN) of the first module on the remote bus branch, either via a line or via a variable. For the hardware, the type of connection corresponds to the INTERBUS cable between two INTERBUS modules.

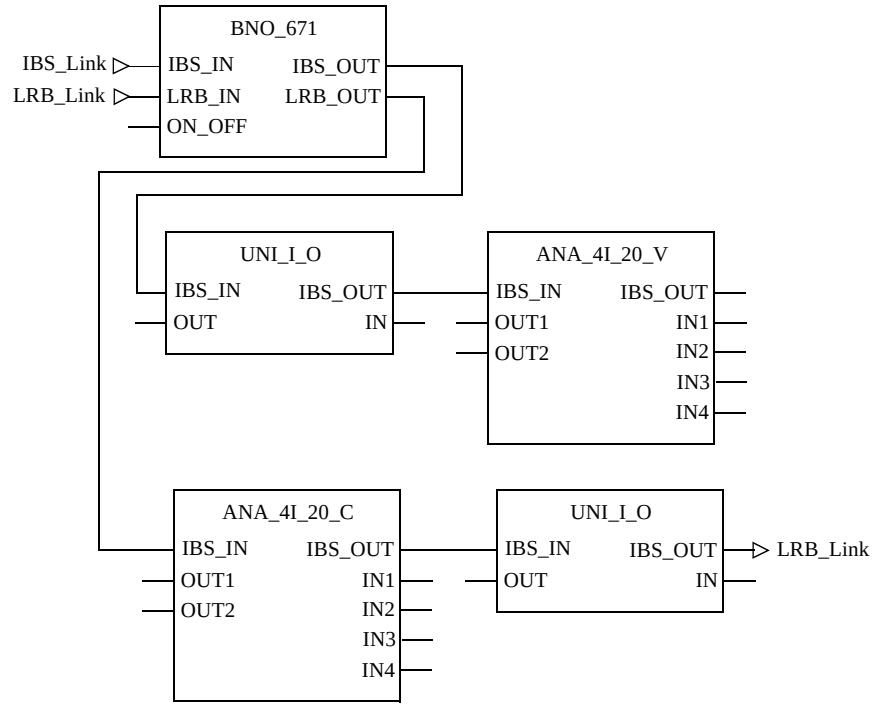
Structure with BNO_671

A possible INTERBUS structure with the bus terminal module is illustrated.

Structure with BNO_671 - Part 1



Structure with BNO_671 - Part 2



COMPACT: Configuring a main rack

30

Overview

Introduction

This chapter describes the block COMPACT.

What's in this chapter?

This chapter contains the following topics:

Brief description

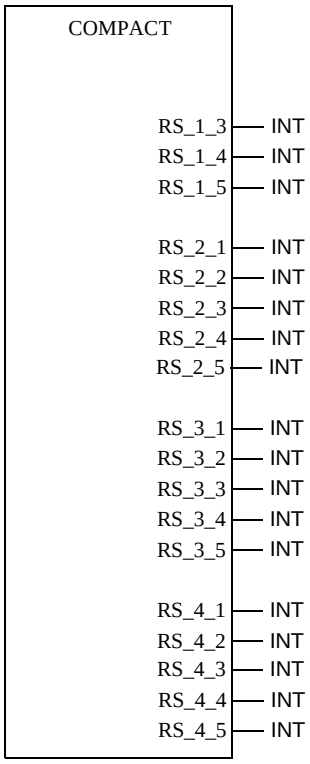
Function description

The Function block is used to edit the configuration data of a Compact primary backplane for subsequent use by the scaling EFBs.
To configure a COMPACT primary backplane, the COMPACT function block is inserted into the configuration section. The function block for configuring the analog module is connected to the SLOT outputs.
EN and ENO can be configured as additional parameters.

Representation

Symbol

Block representation:



Parameter description

Block parameter description:

Parameter	Data type	Meaning
RS_1_3	INT	Rack 1, slot 3
RS_1_4	INT	Rack 1, slot 4
RS_1_5	INT	Rack 1, slot 5
RS_2_1	INT	Rack 2, slot 1
:	:	:
RS_2_5	INT	Rack 2, slot 5
RS_3_1	INT	Rack 3, slot 1
:	:	:
RS_3_5	INT	Rack 3, slot 5
RS_4_1	INT	Rack 4, slot 1
:	:	:
RS_4_5	INT	Rack 4, slot 5

Runtime error

Runtime error

Internal I/O map errors will cause an Error message.

DAU202: Configuring the Compact module DAU 202 / DAU 252 / DAU 282

31

Overview

Introduction

This chapter describes the block DAU202.

What's in this chapter?

This chapter contains the following topics:

Brief description

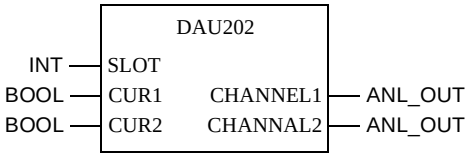
Function description

The function block is used to edit the configuration data of a DAU 202 / DAU 252 / DAU 282 Compact module for subsequent use by the scaling EFBs.
This module has two bipolar output channels for mixed voltage and current processing.
For the configuration of a DAU 202 / DAU 252 / DAU 282, the function block in the Configuration section will be connected to the corresponding SLOT output of the COMPACT Function block. The 3x references and 4x references specified in the I/O map are automatically assigned internally to the individual channels and can therefore only be occupied by Unlocated variables.
The analog values can be further processed in Scaling sections using the O_DEBUG, O_NORM, O_SCALE, O_PHYS, O_DBSET, O_RAW Function blocks. EN and ENO can be configured as additional parameters.

Representation

Symbol

Block representation:



Parameter description

Block parameter description:

Parameter	Data type	Meaning
SLOT	INT	Slot
CUR1	BOOL	0: Channel 1 processing voltage 1: Channel 1 processing current
CUR2	BOOL	0: Channel 2 processing voltage 1: Channel 2 processing current
CHANNEL1	ANL_OUT	Channel 1
CHANNEL2	ANL_OUT	Channel 2

Runtime error

Runtime error

If no DAU 202 / DAU 252 / DAU 282 module has been configured for the specified SLOT input, an error message will appear.
Status information cannot be requested for this module.

DAU204: Configuring the Compact module DAU 204

32

Overview

Introduction

This chapter describes the block DAU204.

What's in this chapter?

This chapter contains the following topics:

Brief description

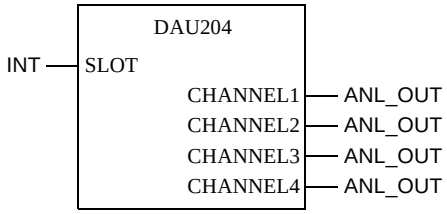
Function description

The function block is used to edit the configuration data of a DAU 204 Compact module for subsequent use by the scaling EFBs.
This module has four output channels for combined bipolar and unipolar voltage and current processing.
For the configuration of an DAU 204, the function block in the Configuration section is connected to the corresponding SLOT output of the COMPACT Function block.
The 4x references specified in the I/O map (as well as the 3 references for the status information) are automatically assigned internally to the individual channels and can therefore only be occupied by Unlocated variables.
The analog values can be further processed in Scaling sections using the O_DEBUG, O_NORM, O_SCALE, O_PHYS, O_DBSET, O_RAW function blocks.
EN and ENO can be configured as additional parameters.

Representation

Symbol

Block representation:



Parameter description

Block parameter description:

Parameter	Data type	Meaning
SLOT	INT	Slot
CHANNEL1	ANL_OUT	Channel 1
CHANNEL2	ANL_OUT	Channel 2
CHANNEL3	ANL_OUT	Channel 3
CHANNEL4	ANL_OUT	Channel 4

Runtime error

Runtime error

If no DAU 204 module has been configured for the specified SLOT input, an error message appears.
A range warning for the channels is not provided by the processing Function blocks (See *Function description, p. 162*).
The status information "Open circuit on channel" and "Open circuit on one or several channels" can be collected via the status register (3x reference) defined in the I/O map.

DAU208: Configuring the Compact module DAU 208

33

Overview

Introduction

This chapter describes the block DAU208.

What's in this chapter?

This chapter contains the following topics:

Brief description

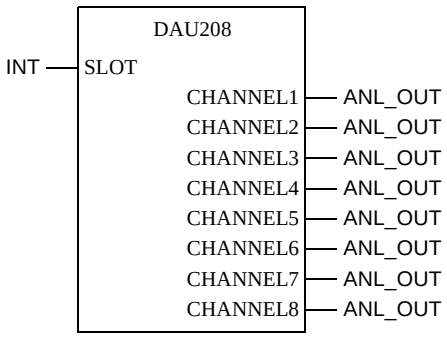
Function description

The function block is used to edit the configuration data of a DAU 208 Compact module for subsequent use by the scaling EFBs.
This module has eight bipolar output channels for voltage processing.
For the configuration of an DAU 208, the function block in the Configuration section is connected to the corresponding SLOT output of the COMPACT Function block.
The 3x references and 4x references specified in the I/O map are automatically assigned internally to the individual channels and can therefore only be occupied by Unlocated variables.
The analog values can be further processed in Scaling sections using the O_DEBUG, O_NORM, O_SCALE, O_PHYS, O_DBSET, O_RAW function blocks.
EN and ENO can be configured as additional parameters.

Representation

Symbol

Block representation:



Parameter description

Block parameter description:

Parameter	Data type	Meaning
SLOT	INT	Slot
CHANNEL1	ANL_OUT	Channel 1
CHANNEL2	ANL_OUT	Channel 2
CHANNEL3	ANL_OUT	Channel 3
CHANNEL4	ANL_OUT	Channel 4
CHANNEL5	ANL_OUT	Channel 5
CHANNEL6	ANL_OUT	Channel 6
CHANNEL7	ANL_OUT	Channel 7
CHANNEL8	ANL_OUT	Channel 8

Runtime error

Runtime error

If no DAU 208 module has been configured for the specified SLOT input, an error message appears.
Status information cannot be requested for this module.

DIG_16I: Configuring the TIO-module BDI 346 00 / 546 50 / 746 50

34

Overview

Introduction

This chapter describes the block DIG_16I.

What's in this chapter?

This chapter contains the following topics:

Brief description

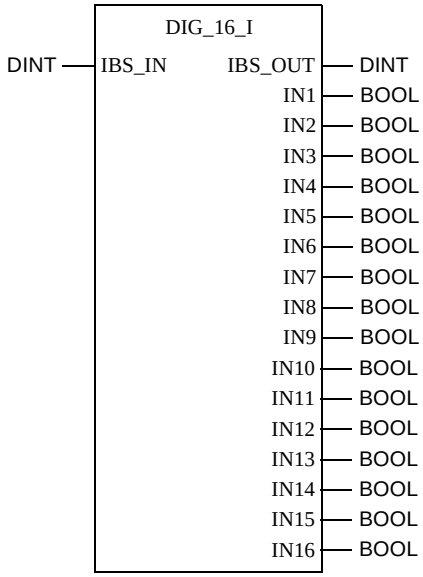
Function description

The DIG_16I Function block is a software connection from INTERBUS hardware modules with 16 binary inputs.
The function block corresponds to the hardware modules TIO/IS 170 BDI 346 00, TIO/IS 170 BDI 546 50 and TIO/IS 170 BDI 746 50. A different designation to that of the hardware was selected in order to clarify the relationship between the name and the function of the module.
EN and ENO can be projected as additional parameters.

Representation

Symbol

Block representation:



Parameter description

Block parameter description:

Parameter	Data type	Meaning
IBS_IN	DINT	Incoming INTERBUS
IBS_OUT	DINT	Outgoing INTERBUS
IN1	BOOL	Input 1 of the TIO
IN2	BOOL	Input 2 of the TIO
:	:	:
IN16	BOOL	Input 16 of the TIO

Detailed description

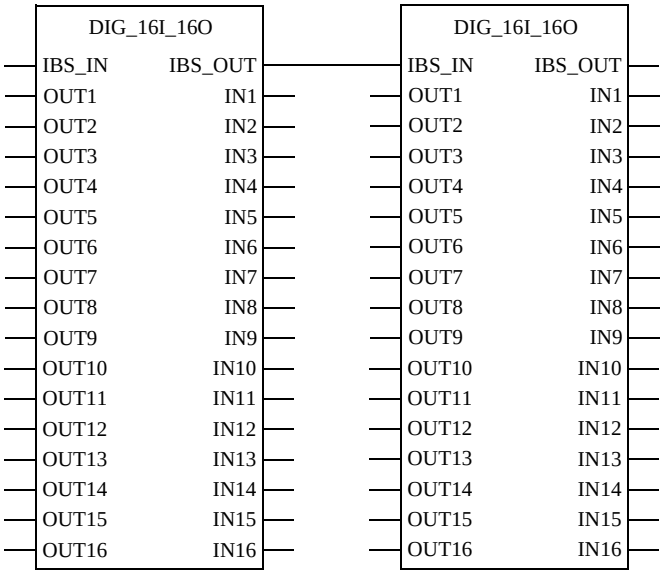
Detailed description

The function block occupies one input word in the Status-RAM.

Parameter description - inputs

IBS_IN

IBS_IN = Connection for the incoming remote bus part of INTERBUS
On the hardware, the male connector is on the top left of the module.
The module is connected to the outgoing master (1st module on the bus) remote bus (IBS_OUT) or the previous module (see also diagram). The link can be made via a line or via a variable. For the hardware, the type of connection corresponds to the INTERBUS cable between two bus devices.
Connection of two INTERBUS modules



Parameter description - Outputs

IBS_OUT	IBS_OUT = Connection for the outgoing remote bus part of INTERBUS On the hardware, the male connector is on the top right of the module. The module is connected to the incoming remote bus (IBS_IN) of the following module, either via a line or via a variable. For the hardware, the type of connection corresponds to the INTERBUS cable between two INTERBUS modules.
INx	INx = Input x x stands for the digit 1 to 16 which indicates the corresponding input. Binary process values are read into the INTERBUS module via the relevant input (INx).

DIG_16I_12O_MON: Configuring the module ADM 390 10

35

Overview

Introduction

This chapter describes the block DIG_16I_12O_MON.

What's in this chapter?

This chapter contains the following topics:

Brief description

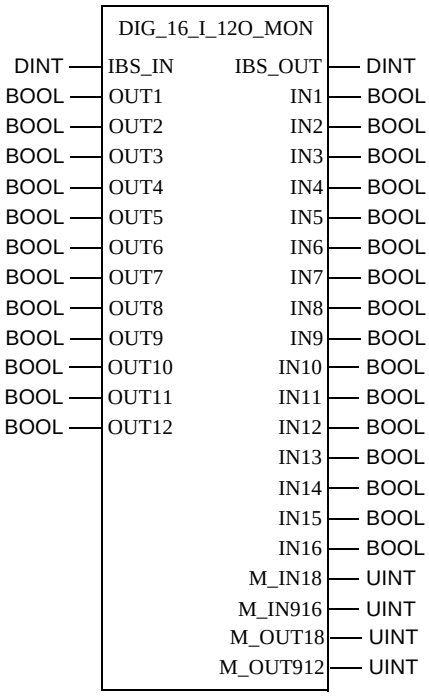
Function description

The DIG_16I_12O_MON Function block is a software connection to INTERBUS Momentum/IS 170 ADM 390 10 hardware modules.
The function block has 16 binary inputs and 12 binary outputs, which can be operated simultaneously or as inputs or outputs only. The status of the I/O is also indicated.
EN and ENO can be projected as additional parameters.

Representation

Symbol

Block representation:



Parameter description

Block parameter description:

Parameters	Data type	Meaning
IBS_IN	DINT	Incoming INTERBUS
OUT1	BOOL	Output 1 of the module
:	:	:
OUT12	BOOL	Output 12 of the module
IBS_OUT	DINT	Outgoing INTERBUS
IN1	BOOL	Input 1 of the module
:	:	:
IN16	BOOL	Input 16 of the module
M_IN18	UDINT	Status indicator for inputs 1 to 8
M_IN916	UDINT	Status indicator for inputs 9 to 16
M_OUT18	UDINT	Status indicator for outputs 1 to 8
M_OUT912	UDINT	Status indicator for outputs 9 to 12

Detailed description

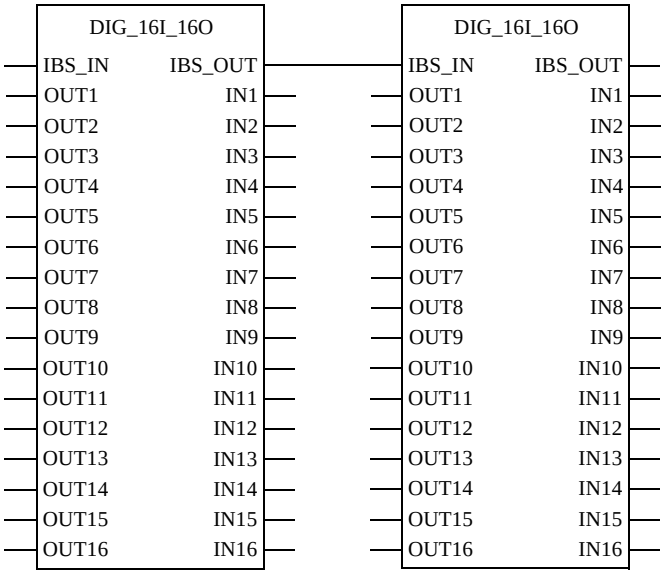
Detailed description

The function block occupies 3 input words and 3 output words in the Status-RAM.

Parameter description - inputs

IBS_IN

IBS_IN = Connection for the incoming remote bus part of INTERBUS
On the hardware, the male connector is on the top left of the module.
The module is connected here to the outgoing remote bus (IBS_OUT) of the master (1st module on the bus) or the preceding module (see diagram). The link can be made via a line or via a variable. For the hardware, the type of connection corresponds to the INTERBUS cable between two bus devices.
Connection of two INTERBUS modules



OUTx = Output x

OUTx = Output x
x stands for the digit 1 to 16 which indicates the corresponding output.
The binary values to be produced via the INTERBUS module are supplied to the process via the relevant output (OUTx).

Parameter description - Outputs

IBS_OUT	IBS_OUT = Connection for the outgoing remote bus part of INTERBUS On the hardware, the male connector is on the top right of the module. The module is connected to the incoming remote bus (IBS_IN) of the following module, either via a line or via a variable. For the hardware, the type of connection corresponds to the INTERBUS cable between two INTERBUS modules.
INx	INx = Input x x stands for the digit 1 to 16 which indicates the corresponding input. Binary process values are read into the INTERBUS module via the relevant input (INx).
M_IN18	M_IN18 = Status indicator for inputs 1 to 8 A number is indicated which refers to the faulty input. Example: M_IN18 = 10001, then input 1 and input 5 are faulty. They are counted from right to left.
M_IN916	M_IN916 = Status indicator for inputs 9 to 16 A number is indicated which refers to the faulty input. Example: M_IN916 = 1001, then input 9 and input 12 are faulty. They are counted from right to left.
M_OUT18	M_OUT18 = Status indicators for outputs 1 to 8 A number is indicated which refers to the faulty output. Example: M_OUT18 = 10001, then output 1 and output 5 are faulty. They are counted from right to left.
M_OUT912	M_OUT912 = Status indicators for outputs 9 to 12 A number is indicated which refers to the faulty output. Example: M_OUT912 = 1001, then output 9 and output 12 are faulty. They are counted from right to left.

DIG_16I_16O: Configuring the TIO-module BDM 346 00

36

Overview

Introduction

This chapter describes the block DIG_16I_16O.

What's in this chapter?

This chapter contains the following topics:

Brief description

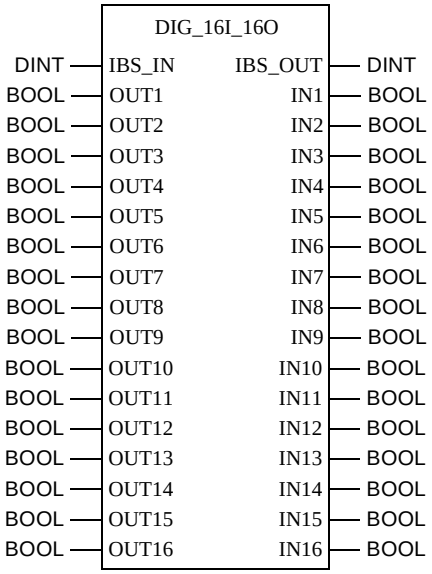
Function description

The DIG_16I_16O Function block in Concept functions in the same way as its hardware counterpart.
However, its operation has been simplified by programming it as a function block in Concept. The module occupies one input word and one output word in the master. EN and ENO can be projected as additional parameters.

Representation

Symbol

Block representation:



Parameter description

Block parameter description:

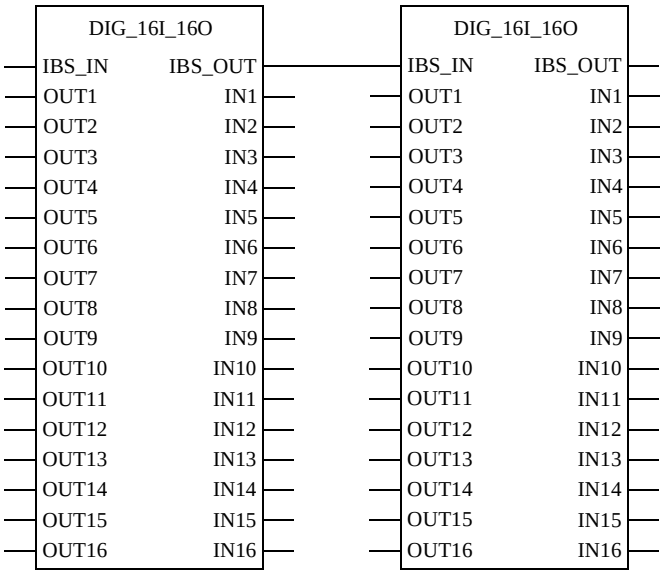
Parameter	Data type	Meaning
IBS_IN	DINT	Incoming INTERBUS
OUT1	BOOL	Output 1 of the TIO
OUT2	BOOL	Output 2 of the TIO
:	:	:
OUT16	BOOL	Output 16 of the TIO
IBS_OUT	DINT	Outgoing INTERBUS
IN1	BOOL	Input 1 of the TIO
IN2	BOOL	Input 2 of the TIO
:	:	:
IN16	BOOL	Input 16 of the TIO

Detailed description

Parameter description - inputs

IBS_IN

IBS_IN = Connection for the incoming remote bus part of INTERBUS
On the hardware, the male connector is on the top left of the module.
The module is connected here to the outgoing remote bus (IBS_OUT) of the master (1st module on the bus) or the preceding module (see diagram). The link can be made via a line or via a variable. For the hardware, the type of connection corresponds to the INTERBUS cable between two bus devices.
Connection of two INTERBUS modules



OUTx = Output x	OUTx = Output x x stands for the digit 1 to 16 which indicates the corresponding output. The binary values to be produced via the INTERBUS module are supplied to the process via the relevant output (OUTx).
Parameter description - Outputs	
IBS_OUT	IBS_OUT = Connection for the outgoing remote bus part of INTERBUS On the hardware, the male connector is on the top right of the module. The module is connected to the incoming remote bus (IBS_IN) of the following module, either via a line or via a variable. For the hardware, the type of connection corresponds to the INTERBUS cable between two INTERBUS modules.
INx	INx = Input x x stands for the digit 1 to 16 which indicates the corresponding input. Binary process values are read into the INTERBUS module via the relevant input (INx).

DIG_16O: Configuring the TIO-modules BDO 346 00 / BDO 946 50

37

Overview

Introduction

This chapter describes the block DIG_16O.

What's in this chapter?

This chapter contains the following topics:

Brief description

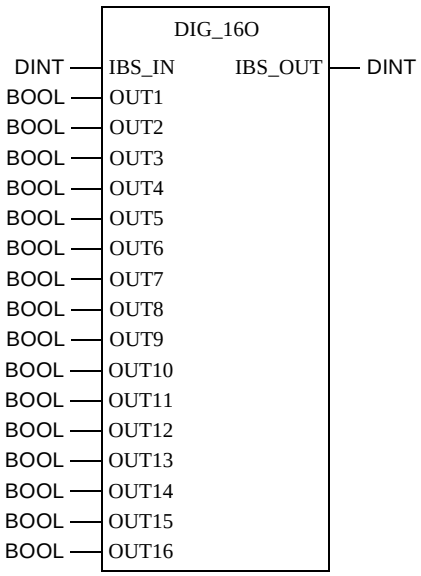
Function description

The DIG_16O Function block is a software connection to INTERBUS hardware modules with 16 binary outputs.
The function block corresponds to the hardware TIO/IS 170 BDO 346 00 and TIO/IS 170 BDO 946 50. A different designation was selected in order to achieve a clearer relation between the name and the function of the module.
EN and ENO can be projected as additional parameters.

Representation

Symbol

Block representation:



Parameter description

Block parameter description:

Parameter	Data type	Meaning
IBS_IN	DINT	Incoming INTERBUS
OUT1	BOOL	Output 1 of the TIO
OUT2	BOOL	Output 2 of the TIO
:	:	:
OUT16	BOOL	Output 16 of the TIO
IBS_OUT	DINT	Outgoing INTERBUS

Detailed description

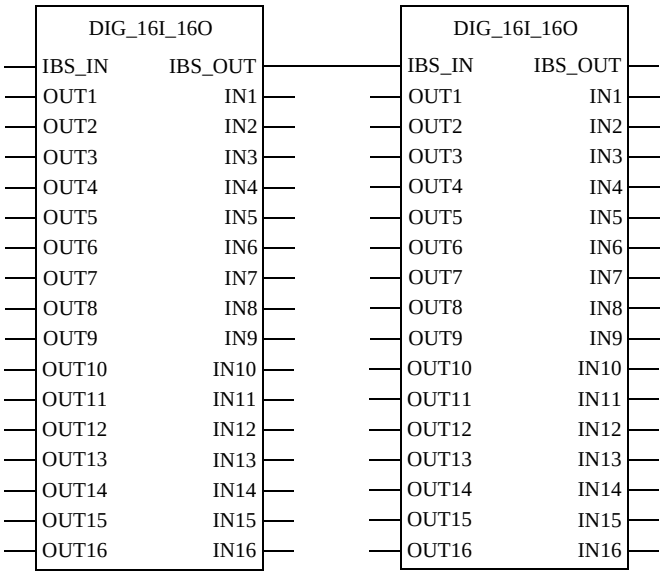
Detailed description

The function block occupies one output word in the Status-RAM.

Parameter description - inputs

IBS_IN

IBS_IN = Connection for the incoming remote bus part of INTERBUS
On the hardware, the male connector is on the top left of the module.
The module is connected here to the outgoing remote bus (IBS_OUT) of the master (1st module on the bus) or the preceding module (see diagram). The link can be made via a line or via a variable. For the hardware, the type of connection corresponds to the INTERBUS cable between two bus devices.
Connection of two INTERBUS modules



OUTx

OUTx = Output x
x stands for the digit 1 to 16 which indicates the corresponding output.
The binary values to be produced via the INTERBUS module are supplied to the process via the relevant output (OUTx).

Parameter description - Outputs

IBS_OUT	IBS_OUT = Connection for the outgoing remote bus part of INTERBUS On the hardware, the male connector is on the top right of the module. The module is connected to the incoming remote bus (IBS_IN) of the following module, either via a line or via a variable. For the hardware, the type of connection corresponds to the INTERBUS cable between two INTERBUS modules.

DROP: Configuring a I/O Station subrack

38

Overview

At a Glance

This chapter describes the block DROP.

What's in this chapter?

This chapter contains the following topics:

Brief description

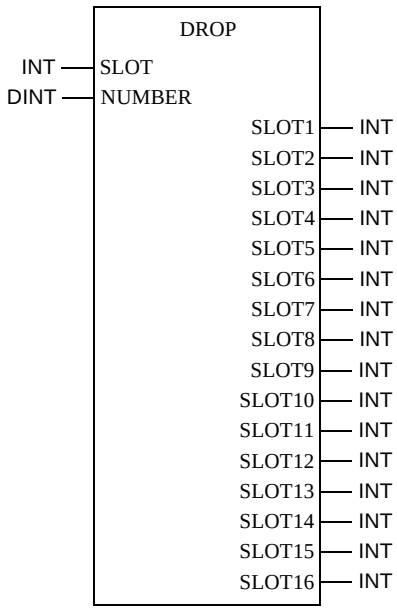
Function description

The Function block is used to edit the configuration data of a remote or distributed I/O station for subsequent processing by module configuration EFBs.
To configure an I/O station subrack, the DROP Function block in the configuration section is connected to the corresponding SLOT output of the QUANTUM Function block. The number of the I/O station defined in the I/O map has to be entered at the NUMBER input of the DROP Function block. The Function blocks for configuration of the analog modules of the I/O stations are connected to the SLOT outputs.
EN and ENO can be projected as additional parameters.

Representation

Symbol

Block representation:



Parameter description

Block parameter description:

Parameters	Data type	Meaning
SLOT	INT	Slot for RIO, DIO, NOM
NUMBER	DINT	Number of RIO, DIO, NOM
SLOT1	INT	Slot 1
:	:	:
SLOT16	INT	Slot 16

Runtime error

Runtime error

If no "Head" has been configured for the I/O station subrack, an error message appears.

Index

A

B

C

D

F

I

L

M

N

O

P

Q

R

S

T

U

W

X