Function block library

AnalogTechnology_10

for PLCnext Engineer

Documentation for
PHOENIX CONTACT function blocks
PHOENIX CONTACT GmbH Co. KG
Flachsmarktstrasse 8
D-32825 Blomberg, Germany

This documentation is available in English only.

Table of Contents

1 Installation hint

If you did not specify a different directory during **library** installation all data in the MSI file will be unpacked to

c:\Users\Public\Documents\Phoenix Contact Libraries\PLCnext Engineer (former: PC Worx Engineer)

Please copy the library data to your PLCnext Engineer (former: PC Worx Engineer) working library directory.

If you did not specify a different directory during **PLCnext Engineer** installation the default PLCnext Engineer working library directory is

c:\Users\Public\Documents\PLCnext Engineer\Libraries (former: PC Worx Engineer\Libraries)

2 General information

The AnalogTechnology library offers function blocks for acquisition and evaluation of analog signals.

3 Change notes

Library version	Library build	PLCnext Engineer version	Change notes	Supported PLCs
10	20200206	>= 2020.0 LTS	Released for 2020.0 LTS	AXC F 1152 (1151412) AXC F 2152 (2404267)
10	20191001	2019.0 LTS 2019.3 2019.6 2019.9	Adapted to 2019.9	AXC F 2152 (2404267)
9	20190920	2019.0 LTS 2019.3 2019.6	AnalogTechnology_9: • Restricted visible function block	AXC F 2152 (2404267)
9	20190828	2019.0 LTS 2019.3 2019.6	AnalogTechnology_9: • ANL_AO_NORM_3: Bug fix for iModuleType = 7	AXC F 2152 (2404267)
8	20190730	2019.0 LTS 2019.3 2019.6	AnalogTechnology_8: • ANL_IL_AO4_8_1: New (converted from PC Worx 6).	AXC F 2152 (2404267)
7	20190718	2019.0 LTS 2019.3 2019.6	Adapted to 2019.6	AXC F 2152 (2404267)
6	20190225	2019.0 LTS	Supports "Allow extended identifiers" = ON	AXC F 2152 (2404267)
6	20190219	2019.0 LTS	AnalogTechnology_6: • Adapted to 2019.0 LTS	AXC F 2152 (2404267)
5	20181115	7.2.3	AnalogTechnology_5: • ANL_IL_AI_4_UI_1: New (converted from PC Worx 6).	AXC F 2152 (2404267)

4	20180913	7.2.3	AnalogTechnology_4: • AI_NORM_3 renamed to ANL_AI_NORM_3. • AO_NORM_2 renamed to ANL_AO_NORM_2. • ANL_IL_AI_2_SF_1: New (converted from PC Worx 6). • ANL_IL_AI_4_EF_1: New (converted from PC Worx 6). • ANL_IL_AI_8_1: New (converted from PC Worx 6). • ANL_IL_AO_2_UI_1: New (converted from PC Worx 6).	AXC F 1050 (2404701) AXC F 2152 (2404267)
3		7.2.3	AnalogTechnology_3: • Does not need PCWEngineerAdaption any more.	AXC F 1050 (2404701) AXC F 2152 (2404267)
2		7.2.3	AnalogTechnology_2: • AI_NORM_2: Bug fix for .SEM1133 Assignments to read-only variables are not valid	AXC F 1050 (2404701) AXC F 2152 (2404267)
1		7.2.1	Converted from PC Worx 6	AXC F 1050 (2404701) AXC F 2152 (2404267)

New version number: Functional changes of at least one function block

New build number: No functional changes, but changes in the MSI file (e.g. documentation update, additional examples)

4 Function blocks

Function block	Description	Version	Supported articles	License
ANL_AI_NORM	Standardization of analog input values for analog modules.	3	IBS RT 24 AIO 4/2-T (?) IBS RT 24 AI 8-T (2723194) IB IL AI 2/SF (2726285) IB ST 24 AI 4/SF-WT (2752534) IB ST 24 BAI 8/I (2721028) IB ST 24 BAI 8/U (2721015) IB ST ZF 24 AI 4/BP (2724737) IB ST ZF 24 AI 4/I (2721264) IB ST ZF 24 AI 4/SF (2750620) IB ST ZF 24 AI 4/SF4 (2750594) IB ST ZF 24 BAI 2/BP (2724957) IB ST ZF 24 BAI 2/SF (2723958) AXL F AI2 AO2 1H (2702072) AXL F AI2 AO2 XC 1H (1035429) AXL F AI4 I 1H (2688491) AXL F AI4 I XC 1H (2702007) AXL F AI4 U 1H (2688501) AXL F AI4 U XC 1H (2702008) AXL F AI8 1F (2688064) AXL F AI8 XC 1F (2701232) AXL F AI8 W 1F (2702525)	none

ANL_AO_NORM	Standardization of analog output values for analog modules.	3	IB IL AO 1/U/SF (2727776) IB IL AO 1/U/SF-PAC (2861399) IB ST 24 AO 4/BP (2752521) IB ST 24 AO 4/SF (2754312) IB ST 24 AO 4/SF/4 (2750578) IB ST 24 AO 4/EF (2700839) IB ST 24 BAO 8/U-8B (2721031) IB ST ZF 24 AO 4/BP (2750617) IB ST ZF 24 AO 4/SF (2750604) IB ST ZF 24 AO 4/SF4 (2750581) IB ST ZF 24 BAO 8/U-8B (2721248) IB ST ZF 24 BAO 8/U (2721251) IBS RT 24 AO 4-T (2723181) IBS RT 24 AO 4-T (2723181) IBSL BOX AO 1/2/I M12 (2723398) IBSL BOX AO 1/2/U M12 (2724025) AXL F AI2 AO2 1H (2702072) AXL F AI2 AO2 XC 1H (1035429) AXL F AO4 1H (2688527) AXL F AO4 XC 1H (2702153) AXL F AO8 1F (2688080) AXL F AO8 XC 1F (2701237)	none
ANL_IL_AI_2_SF	Parameterization and control.	1	IB IL AI 2/SF (2726285) IB IL AI 2/SF-PAC (2861302)	none
ANL_IL_AI_4_EF	Parameterization and control.	1	IB IL AI 4/EF (2863478) IB IL AI 4/EF-PAC (2878447) IB IL AI 4/EF-2MBD (2878544) IB IL AI 4/EF-2MBD-PAC (2878641)	none
ANL_IL_AI_8	Parameterization and standardization.	1	IB IL AI 8/SF (2727831) IB IL AI 8/SF-PAC (2861412) Only for current range: IB IL AI 8/IS (2742748) IB IL AI 8/IS-PAC (2861661)	none

ANL_IL_AO_1_SF	Parameterization and control	1	IB IL AO 1/SF-PAC (2861315)	none
ANL_IL_AO_2	Parameterization and operation.	1	IB IL AO 2 /U/BP-PAC (2861467) IB IL AO 2 /SF-PAC (2863083)	none
ANL_IL_AO_2_UI	Parameterization and control.	1	IB IL AO 2/UI-PAC (2700775)	none
ANL_IL_AI_4_UI	Parameterization and standardization.	1	IB IL AI 4 /I-PAC (2700458) IB IL AI 4 /U-PAC (2700459)	none
ANL_IL_AO4_8	Parameterization and standardization.	1	IB IL AO 2 /U/BP-PAC (2861467) IB IL AO 2 /U/BP (2732732) IB IL AO 4/8 /U/BP 2MBD-PAC (2878052)	none

5 ANL_AI_NORM

The ANL_AI_NORM fubction block cyclically converts the analog values of the Phoenix Contact analog modules in a measuring range preset by the user (rLoLim to rHiLim).

Caution: The function block does not support 16-bit representation of hardware error messages (iModuleType = 3, 6, 7).

The function block is not responsible for hardware error messages (module) or for parameterizing the module.

Example

- Measuring range : 0 V to 10 V (unipolar)
- Parameterized rLoLim : 2000
- Parameterized rHiLim : 4000
- Applied analog value at

  - wln ->Output at rOUT
  - 0 V -> 2000
  - 5 V -> 3000
  - 10 V -> 4000

5.1 Function block call
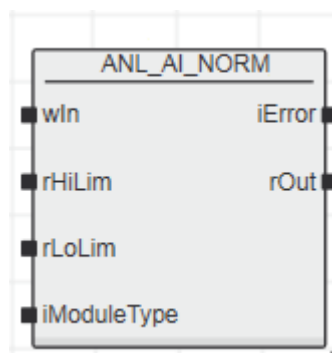
5.2 Input parameters

Name	Type	Description
wln	WORD	Analog data from module
rHiLim	REAL	Upper limit value of the standardized range
rLoLim	REAL	Lower limit value of the standardized range
iModuleType	INT	Module type

5.3 Output parameters

Name	Type	Description

iError	INT	
		• 0 = No error • 1 = Invalid module type • 2 = Limit value error: rHiLim <= rLoLim • 3 = Overrange • 4 = Open circuit (not for iModuleType = 3, 6, 7)
rOut	REAL	Standardized value linear between rLoLim and rHiLim

Caution: In case of error, the last valid output values remain unchanged in status.

5.4 Diagnosis

This function block has no diagnosis.

5.5 Module types

5.5.1 Axioline

Module Type		Range	MODUL_TYPE
AXL F AI2 AO2 1H (2702072) AXL F AI2 AO2 XC 1H (1035429)	IL	0 - 10 V 0 - 5 V 0 - 20 mA 4 - 20 mA	7
		+/- 10 V +/- 5V +/- 20 mA	8
	S7	0 - 10 V 0 - 5 V 0 - 20 mA	1
		+/- 10 V +/- 5 V +/- 20 mA	4
		4 - 20 mA	2
AXL F AI4 I 1H (2688491) AXL F AI4 I XC 1H (2702007)	IL	0 - 20 mA 4 - 20 mA	7
		+/- 20 mA	8
	S7	0 - 20 mA	1
		+/- 20 mA	4
		4 - 20 mA	2
AXL F AI4 U 1H (2688501) AXL F AI4 U XC 1H (2702008)	IL	0 - 10 V	7
		+/- 10 V	8
	S7	0 - 10 V	1
		+/- 10 V	4
AXL F AI8 1F (2688064) AXL F AI8 XC 1F (2701232) AXL F AI8 W 1F (2702525)	IL	0 - 10 V 0 - 5 V 0 - 20 mA 4 - 20 mA	7
		+/- 10 V +/- 5V +/- 20 mA	8
	S7	0 - 10 V 0 - 5 V 0 - 20 mA	1
		+/- 10 V +/- 5V +/- 20 mA	4
		4 - 20 mA	2

5.5.2 Inline

Module Type		Data width	Polarity	Range	MODUL_TYPE
IB IL AI 2/SF (2726285)	ST	12 bits	unipolar	0-10 V, 0-20 mA	1
.	ST	12 bits	unipolar	4-20 mA	2
.	ST	12 bits	bipolar	+/-10 V, +/-20 mA	4
.	RT	16 bits	unipolar	0-10 V, 0-20 mA	3
.	RT	16 bits	unipolar	4-20 mA	2
.	RT	16 bits	bipolar	+/-10 V, +/-20 mA	6
.	IL	16 bits	unipolar	0-10 V, 0-20 mA, 4-20 mA	7
.	IL	16 bits	unipolar	+/-10 V, +/-20 mA	8
IB ST ZF 24 AI 4/BP (2724737)	.	12 bits	unipolar	4-20 mA	2
.	.	12 bits	bipolar	+/-10 V	4
IB ST ZF 24 AI 4/I (2721264)	.	12 bits	unipolar	0-10 V, 0-20 mA	1
.	.	12 bits	unipolar	2-10 V, 4-20 mA	2
IB ST ZF 24 AI 4/SF (2750620)	.	12 bits	unipolar	0-10 V, 0-20 mA	1
IB ST ZF 24 AI 4/SF4 (2750594)	.	12 bits	unipolar	0-10 V, 4-20 mA	1
IB ST 24 AI 4/SF-WT (2752534)	.	12 bits	unipolar	0-10 V, 0-20 mA	1
IB ST ZF 24 BAI 2/BP (2724957)	.	12 bits	bipolar	+/- 10 V, +/- 20 mA	4
IB ST ZF 24 BAI 2/SF (2723958)	.	12 bits	unipolar	0-10 V, 0-20 mA	1
IBS RT 24 AI 8-T (2723194)	.	12 bits	unipolar	0-10 V, 0-5 V	1
.	.	12 bits	unipolar	1-5 V, 0-20 mA	1
.	.	12 bits	unipolar	4-20 mA	1
.	.	12 bits	bipolar	+/- 5/10 V, +/- 20 mA	4
IBS RT 24 AIO 4/2-T (?)	.	16 bits	unipolar	0-10V, 0-5V	3
.	.	16 bits	unipolar	0.2-1V, 0-1V	3
.	.	16 bits	unipolar	1-5V, 0-20mA	3
.	.	16 bits	unipolar	4-20mA, 2-10V	3
.	.	16 bits	bipolar	+/- 5/10V, +/- 20mA	6
.	.	16 bits	bipolar	+/- 1V	6
IB ST 24 BAI 8/I (2721028)	.	12 bits	unipolar	0-20mA, 4-20mA	9
IB ST 24 BAI 8/U (2721015)	.	12 bits	unipolar	0-10V, 0-20mA	9

6 ANL_AO_NORM

The ANL_AO_NORM function block converts a standardized value on an analog output module. This module is intended for the listed output modules manufactured by Phoenix Contact.

The ANL_AO_NORM function block cyclically converts the rIn values into analog values for the Phoenix Contact analog modules. The block is not responsible for hardware error messages (module) or for parameterizing the module.

Example Unipolar

rLoLim	rHiLim	rIn	Voltage- or current output in %
0.0	10.0	7.5	75%
0	20	10	50%
-20.0	0.0	-15.0	25%
2000	4000	2000	0%

Example Bipolar

rLoLim	rHiLim	rIn	Voltage- or current output in %
-10.0	10.0	7.5	75%
-10.0	10.0	0.0	0%
-10.0	10.0	-5.0	-50%
0	20	15	50%
2000	4000	2000	-100%

6.1 Function block call

```
        ANL_AO_NORM
  ■ rIn                    iError ■

  ■ rHiLim                  wOut ■

  ■ rLoLim

  ■ iModuleType
```
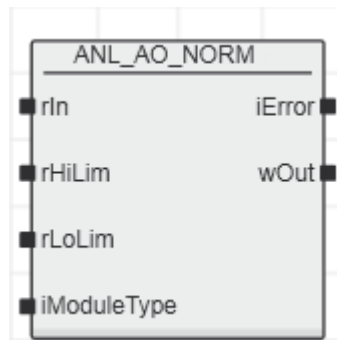
6.2 Input parameters

Name	Type	Description
rIn	REAL	Analog value to be output (value in the range between rLoLim and rHiLim)
rHiLim	REAL	Upper limit value of the standardized range
rLoLim	REAL	Lower limit value of the standardized range
iModuleType	INT	Module type. Please refer to list below.

6.3 Output parameters

Name	Type	Description
iError	INT	0 = No error 1 = Invalid module type 3 = Limit value error: rHiLim is smaller or equal rLoLim 4 = rIn is not in the range between rLoLim and rHiLim
wOut	WORD	Output variable of analog channel

Caution: In case of error, the last valid output values remain unchanged in status.

6.4 Diagnosis

This function block has no diagnosis. Please refer to output parameter "iError".

6.5 Module types

6.5.1 Axioline

Supported Hardware	Data format	Polarity / range	Module type
AXL F AI2 AO2 1H (2702072) AXL F AI2 AO2 XC 1H (1035429) AXL F AO4 1H (2688527) AXL F AO4 XC 1H (2702153) AXL F AO8 1F (2688080) AXL F AO8 XC 1F (2701237)	IL	unipolar: 0 - 10 V 0 - 5 V 0 - 20 mA 4 - 20 mA	7
AXL F AI2 AO2 1H (2702072) AXL F AI2 AO2 XC 1H (1035429) AXL F AO4 1H (2688527) AXL F AO4 XC 1H (2702153) AXL F AO8 1F (2688080) AXL F AO8 XC 1F (2701237)	IL	bipolar: +/- 10 V +/- 5 V	8
AXL F AI2 AO2 1H (2702072) AXL F AI2 AO2 XC 1H (1035429) AXL F AO4 1H (2688527) AXL F AO4 XC 1H (2702153) AXL F AO8 1F (2688080) AXL F AO8 XC 1F (2701237)	S7	unipolar: 0 - 10 V 0 - 5 V 0 - 20 mA 4 - 20 mA	9
AXL F AI2 AO2 1H (2702072) AXL F AI2 AO2 XC 1H (1035429) AXL F AO4 1H (2688527) AXL F AO4 XC 1H (2702153) AXL F AO8 1F (2688080) AXL F AO8 XC 1F (2701237)	S7	bipolar: +/- 10 V +/- 5 V	10

6.5.2 Inline

Module Type	Range	MODUL_TYPE
IB IL AO 1/U/SF (2727776) IB IL AO 1/U/SF-PAC (2861399)	0-10V	3
IB ST 24 AO 4/BP (2752521)	+/- 10V	2
IB ST 24 AO 4/SF (2754312)	0-10 V 0-20 mA	1
IB ST 24 AO 4/SF/4 (2750578)	0-10 V 0-20 mA	1
IB ST 24 AO 4/EF (2700839)	0-10 V 0-20 mA	1
IB ST 24 AO 4/EF (2700839)	4-20 mA	2
IB ST 24 AO 4/EF (2700839)	+/- 10 V	4
IB ST 24 BAO 8/U-8B (2721031)	0-10 V	4
IB ST ZF 24 AO 4/BP (2750617)	+/- 10 V	2
IB ST ZF 24 AO 4/SF (2750604)	0-10 V 0-20 mA	1
IB ST ZF 24 AO 4/SF4 (2750581)	0-10 V 4-20 mA	1
IB ST ZF 24 BAO 8/U-8B (2721248)	0-10 V	4
IB ST ZF 24 BAO 8/U (2721251)	+/- 10V +/- 12 V	2
IBS RT 24 AO 4-T (2723181)	+/- 10V	2 (Voltage)
IBS RT 24 AO 4-T (2723181)	0-20 mA 4-20 mA	1 (Current)
IBSL BOX AO 1/2/I M12 (2723398)	0-20 mA 4-20 mA	3
IBSL BOX AO 1/2/U M12 (2724025)	+/- 10 V	2

7 ANL_IL_AI_2_SF

The ANL_IL_AI_2_SF function standardized an analog value from the IB IL AI 2/SF-PAC (2861302) analog input module between the limits rLO_Limit and rHI_Limit.

Example:

Measuring range	Parameterized rLO_Limit	Parameterized rHI_Limit	Applied analog value	Output rNormValue
0-10V	2000.0	4000.0	0.0	2000.0
0-10V	2000.0	4000.0	5.0	3000.0
0-10V	2000.0	4000.0	10.0	4000.0

The ANL_IL_AI_2_SF function block cyclically converts the analog values of the IB IL AI 2/SF-PAC (2861302) analog Inline module in a measuring range preset by the user (rLO_Limit to rHI_Limit).

The right configuration of the Inline module for the operation will be build by the function block and is defined by the wOutputAddress parameter. The IB IL AI 2/SF-PAC (2861302) provides different formats (IL, ST, RT) and they can be enabled by parameter.

Also the different ranges of the module can be enabled by parameter. The IB IL AI 2/SF-PAC (2861302) module consists of two analog input channels. The channels will be read and configured with process data.

The ANL_IL_AI_2_SF function block can always read or configure only one channel of the module. If you want to use both channels for your application, you have to create two instances of this function block in the PC Worx program.

7.1 Function block call
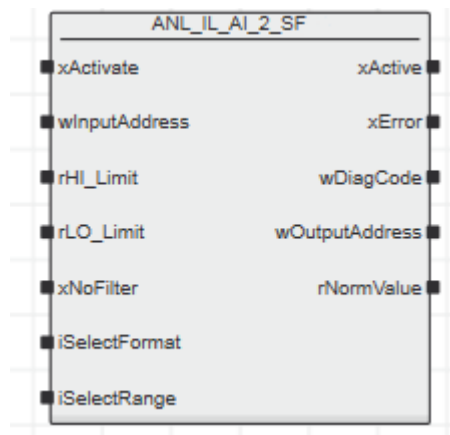
7.2 Input parameters

Name	Type	Description
xActivate	BOOL	Block activation (TRUE = Active)
wInputAddress	WORD	Analog input value
rHI_Limit	REAL	High limit of the normed range
rLO_Limit	REAL	Low limit of the normed range
xNoFilter	BOOL	Disable the filter (In default mode, a filter is activated which generates a measurement of the last 16 values. With this parameter you can disable this filter) -> For more details see datasheet
iSelectFormat	INT	Choose the format: • 0: (IL) IB-IL-Format (15 Bit) • 1: (ST) IB-ST-Format (12 Bit) • 2: (RT) IB-RT-Format (15 Bit) -> For more details see data sheet
iSelectRange	INT	Choose the range: • 0: Unipolar (0-10V) • 1: Unipolar (0-20mA) • 2: Unipolar (4-20mA) • 3: Bipolar (+-10V) • 4: Bipolar (+-20mA) -> For more details see data sheet

7.3 Output parameters

Name	Type	Description
xActive	BOOL	TRUE: Function block is active. FALSE: Function block is not active.
xError	BOOL	Error during initialization or operation.
wDiagCode	WORD	TRUE: An error has occurred. For details refer to wDiagCode and wAddDiagCode.
wOutputAddress	WORD	Controlword for configuration of the inline-clip. (Take care that you connect this parameter with the right channel) -> For more details see the datasheet
rNormValue	REAL	Normed value between rLO_Limit and rHI_Limit

7.4 Diagnosis

wDiagCode	Meaning
16#0000	Block is not activated.
16#8000	Block is active and operating without errors.
16#C001	Error from the Inline clip.
	Measuring range exceeded (overrange).
16#C002	Error from the Inline clip.
	Open circuit.
16#C003	Error from the Inline clip.
	Measured value invalid/no valid measured value available (e.g.,because the channel has not been configured).
16#C004	Error from the Inline clip.
	Configuration faulty.
16#C005	Error from the Inline clip.
	Device faulty
16#C006	Error from the Inline clip.
	Below measuring range (underrange).
16#C010	Error from the function block.
	The desired format is not allowed. Valid values are from 0 to 2.
16#C020	Error from the function block
	The desired format is not allowed. Valid values are from 0 to 4.
16#C030	Error from the function block.
	The Limits are not correct. OGR <= UGR

7.5 Startup instructions

For the startup instruction of the ANL_IL_AI_2_SF function block the following examples are available:

- AnalogTechnology_10_EXA_ANL_IL_AI_2_SF_U.pcwex
- AnalogTechnology_10_EXA_ANL_IL_AI_2_SF_I.pcwex

The examples are located in the "Examples" folder of the unzipped msi-file of the library. They describe the use of the IB IL AI 2/SF (2726285) module and the ANL_IL_AI_2_SF function block with voltage (U) or current (I) with an AXC F 2152 (2404267) controller.

7.5.1 Bus structure

For these examples the following hardware is used:

- IB IL AI 2/SF-PAC (2861302)
- IB IL AO 2/UI-PAC (2700775)

7.5.2 Block connection

Voltage:

- AnalogTechnology_10_EXA_ANL_IL_AI_2_SF_U.pcwex

	ANL_IL_AO_2_UI_Para	
arrParameter	arrParameter	arrParameter
[...]	[...]	
INT#1	1	
INT#1	iChannelNo	
INT#0	0	
INT#0	iRange	
10.0	10.0	
10.0	rHiLim	
0.0	0.0	
0.0	rLoLim	
5.0	5.0	
5.0	rSetValue	

	ANL_IL_AO_2_UI_Para	
arrParameter	arrParameter	arrParameter
[...]	[...]	
INT#2	2	
INT#2	iChannelNo	
INT#1	1	
INT#1	iRange	
10.0	10.0	
10.0	rHiLim	
-10.0	-10.0	
-10.0	rLoLim	
5.0	5.0	
5.0	rSetValue	

ANL_IL_AO_2_UI

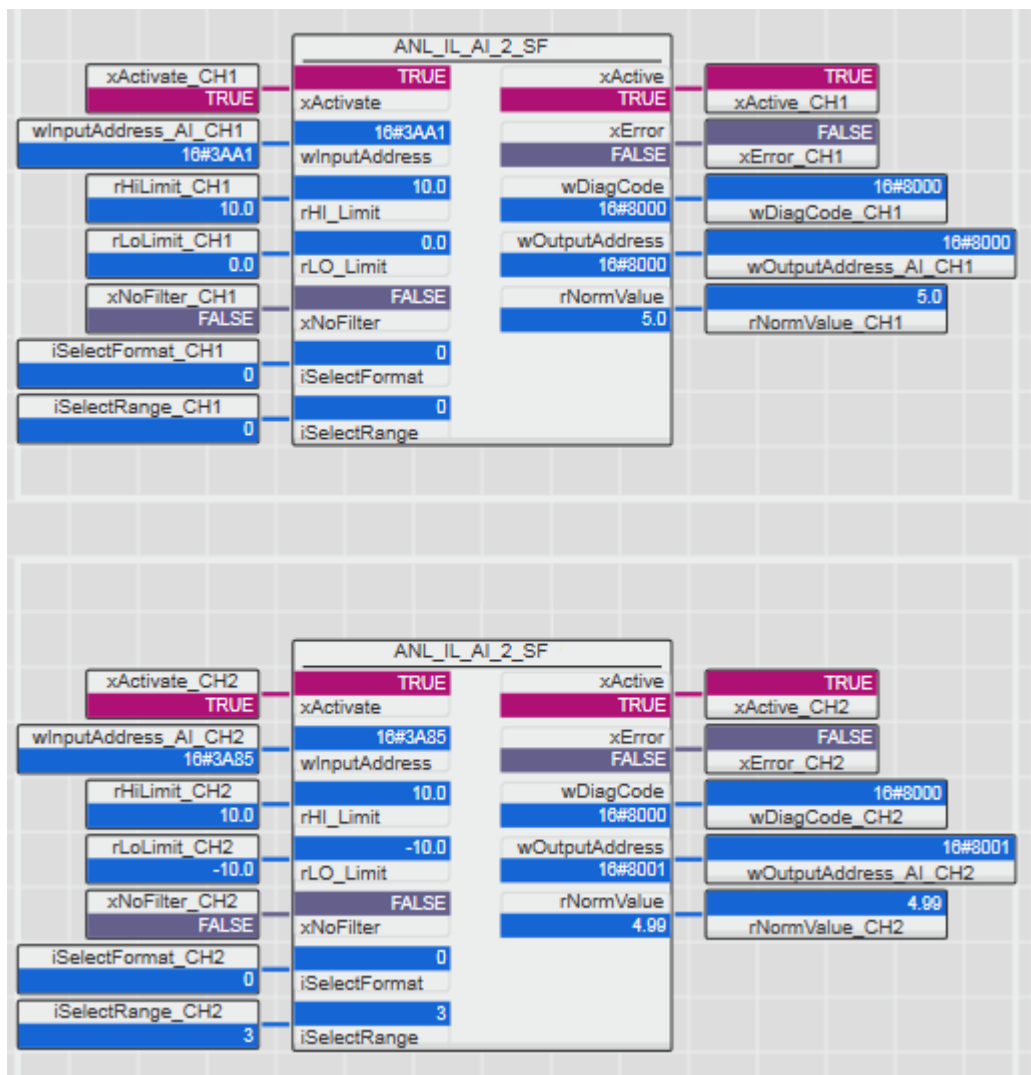|			
xActivate	TRUE	xReady	TRUE
TRUE	xActivate	TRUE	xReady
iSelectCH	3	xError	FALSE
3	iSelectCH	FALSE	xError
arrParameter	[...]	wDiagCode	16#8000
[...]	arrParameter	16#8000	wDiagCode
wInputPD1_AO	16#0000	wAddDiagCode	16#0000
16#0000	wInputPD1	16#0000	wAddDiagCode
wInputPD2_AO	16#0001	arrDiag	[...]
16#0001	wInputPD2	[...]	arrDiag
wInputPD3_AO	16#3A98	wOutputPD1	16#8000
16#3A98	wInputPD3	16#8000	wOutputPD1_AO
wInputPD4_AO	16#3A98	wOutputPD2	16#8001
16#3A98	wInputPD4	16#8001	wOutputPD2_AO
		wOutputPD3	16#3A98
		16#3A98	wOutputPD3_AO
		wOutputPD4	16#3A98
		16#3A98	wOutputPD4_AO

Current:

- AnalogTechnology_10_EXA_ANL_IL_AI_2_SF_I.pcwex

	ANL_IL_AI_2_SF	
xActivate_CH1		xActive
TRUE	TRUE	TRUE
	xActivate	xActive_CH1
wInputAddress_AI_CH1		xError
16#3A95	16#3A95	FALSE
	wInputAddress	xError_CH1
rHiLimit_CH1		wDiagCode
20.0	20.0	16#8000
	rHI_Limit	wDiagCode_CH1
rLoLimit_CH1		wOutputAddress
0.0	0.0	16#8008
	rLO_Limit	wOutputAddress_AI_CH1
xNoFilter_CH1		rNormValue
FALSE	FALSE	10.0
	xNoFilter	rNormValue_CH1
iSelectFormat_CH1		
0	0	
	iSelectFormat	
iSelectRange_CH1		
1	1	
	iSelectRange	

	ANL_IL_AI_2_SF	
xActivate_CH2		xActive
TRUE	TRUE	TRUE
	xActivate	xActive_CH2
wInputAddress_AI_CH2		xError
16#1D25	16#1D23	FALSE
	wInputAddress	xError_CH2
rHiLimit_CH2		wDiagCode
20.0	20.0	16#8000
	rHI_Limit	wDiagCode_CH2
rLoLimit_CH2		wOutputAddress
4.0	4.0	16#800A
	rLO_Limit	wOutputAddress_AI_CH2
xNoFilter_CH2		rNormValue
FALSE	FALSE	7.98
	xNoFilter	rNormValue_CH2
iSelectFormat_CH2		
0	0	
	iSelectFormat	
iSelectRange_CH2		
2	2	
	iSelectRange	

ANL_IL_AO_2_UI_Para

arrParameter	arrParameter	arrParameter
[...]	[...]	

INT#1	1	iChannelNo
INT#1		
INT#2	2	iRange
INT#2		
20.0	20.0	rHiLim
20.0		
0.0	0.0	rLoLim
0.0		
10.0	10.0	rSetValue
10.0		

ANL_IL_AO_2_UI_Para

arrParameter	arrParameter	arrParameter
[...]	[...]	

INT#2	2	iChannelNo
INT#2		
INT#4	4	iRange
INT#4		
20.0	20.0	rHiLim
20.0		
4.0	4.0	rLoLim
4.0		
8.0	8.0	rSetValue
8.0		

ANL_IL_AO_2_UI

xActivate	TRUE	TRUE	xReady	TRUE	xReady
TRUE	xActivate		xError	FALSE	xError
iSelectCH	3		FALSE		
3	iSelectCH		wDiagCode	16#8000	wDiagCode
arrParameter	[...]		16#8000		
[...]	arrParameter		wAddDiagCode	16#0000	wAddDiagCode
wInputPD1_AO	16#0008		16#0000		
16#0008	wInputPD1		arrDiag	[...]	arrDiag
wInputPD2_AO	16#000A		[...]		
16#000A	wInputPD2		wOutputPD1	16#8008	wOutputPD1_AO
wInputPD3_AO	16#3A98		16#8008		
16#3A98	wInputPD3		wOutputPD2	16#800A	wOutputPD2_AO
wInputPD4_AO	16#1D4C		16#800A		
16#1D4C	wInputPD4		wOutputPD3	16#3A98	wOutputPD3_AO
			16#3A98		
			wOutputPD4	16#1D4C	wOutputPD4_AO
			16#1D4C		

8 ANL_IL_AI_4_EF

This function block is used to parameterize the IB IL AI 4/EF (2863478) and IB IL AI 4/EF-PAC (2878447) module. In addition, the block offers detailed diagnostic properties.

Available operating modes:

- Read analog values (default): Analog input values are read in the set format.
- Read configuration: The set configuration of the channels is displayed.
- Read device data: The firmware and device data is displayed.

8.1 Function block call

8.2 Input parameters

Name	Type	Description
xActivate	BOOL	Rising edge: Activates the function block. FALSE: Deactivates the function block.
iSelect	INT	Selection of the operating mode: • 0= Read analog values (default):(Output: wValueChannel1…4) • 1= Read configuration:(Output: wValueChannel1) • 2= Read configuration:(Output: wValueChannel2) • 3= Read configuration: (Output: wValueChannel3) • 4= Read configuration:(Output: wValueChannel4) • 5= Read device data:(Output: wValueChannel1)
dwConfig	DWORD	Kanal 4 / Byte 3 / XX XX XXXX Kanal 3 / Byte 2 / XXXX XXXX Kanal 2 / Byte 1 / XXXX XXXX Kanal 1 / Byte 0 / XXXX XXXX Messbereich (Bit 0-3): 0000 = 0 V bis 10 V 0001 = +/- 10 V 0010 = 0 V bis 5 V 0011 = +/- 5 V 1000 = 0 mA bis 20 mA 1001 = +/- 20 mA 1010 = 4 mA bis 20 mA Format (Bit 4-5): 00 = IB IL 01 = IB ST 10 = S7 Kompatibel 11 = Normiert Filter (Bit 6-7): 00 = Mittelwert 16 01 = Kein Mittelwert 10 = Mittelwert 4 11 = Mittelwert 32 If, for example, all channels are set to 0 mA - 20 mA, ST format, and average value of 32, dwConfig is 0xD8D8D8D8
xReset	BOOL	Resets the block to its initial settings. Errors will be acknowledged.
arrPDU_IN	AIEF_ARR_b_0_9	Input process data item of the IB IL AI 4/EF-PAC (2878447) module.

8.3 Output parameters

Name	Type	Description
xReady	BOOL	The block is ready for operation and working without errors.
xError	BOOL	TRUE: An error has occurred. For details refer to wDiagCode and wAddDiagCode.
wDiagCode	WORD	Diagnosis code. Refer to diagnostic table.
wAddDiagCode	WORD	Additional diagnosis code. Refer to diagnostic table.
wValueChannel1	WORD	Output channel 1.
wValueChannel2	WORD	Output channel 2.
wValueChannel3	WORD	Output channel 3.
wValueChannel4	WORD	Output channel 4.
arrPDU_OUT	AIEF_ARR_b_0_9	Output process data item of the IB IL AI 4/EF-PAC (2878447) module.

8.4 Diagnosis

| wDiagCode	wAddDiagCode	Description
16#0000	16#0000	Function block is deactivated.
16#8000		Function block is in regular operation
16#C100	16#0000	Value of iSelect must be between 0…5.
16#C200	16#0000	No communication with the block. Check process data connection. Device may not be ready for operation.
16#C210	16#0000	Incorrect dwConfig.
	16#xxx1	Incorrect configuration, byte 0 channel 1.
	16#xx1x	Incorrect configuration, byte 1 channel 2.
	16#x1xx	Incorrect configuration, byte 2 channel 3.
	16#1xxx	Incorrect configuration, byte 3 channel 4.
16#C300	16#0000	Range error/error bit.
	16#xxx1	Overrange, channel 1.
	16#xxx2	Underrange, channel 1.
	16#xxx3	Open circuit, channel 1.
	16#xxx4	Measured value invalid, channel 1.
	16#xxx5	Sensor – analog voltage of channel 1 too low.
	16#xxx6	Channel 1 faulty.
	16#xx1x	Overrange, channel 2.
	16#xx2x	Underrange, channel 2.
	16#xx3x	Open circuit, channel 2.
	16#xx4x	Measured value invalid, channel 2.
	16#xx5x	Sensor – analog voltage of channel 2 too low.
	16#xx6x	Channel 2 faulty.
	16#x1xx	Overrange, channel 3.
	16#x2xx	Underrange, channel 3.
	16#x3xx	Open circuit, channel 3.
	16#x4xx	Measured value invalid, channel 3.
	16#x5xx	Sensor – analog voltage of channel 3 too low.
	16#x6xx	Channel 3 faulty.
	16#1xxx	Overrange, channel 4.
	16#2xxx	Underrange, channel 4.
	16#3xxx	Open circuit, channel 4.
	16#4xxx	Measured value invalid, channel 4.
	16#5xxx	Sensor – analog voltage of channel 4 too low.
	16#6xxx	Channel 4 faulty.
16#C400	16#0000	Communication with block aborted/faulty.
16#CFFF	16#0000	Error bit received, cause unknown.

8.5 Startup instructions

For the startup instruction of the ANL_IL_AI_4_EF function block the following example is available:

- AnalogTechnology_10_EXA_ANL_IL_AI_4_EF.pcwex

The example is located in the "Examples" folder of the unzipped msi-file of the library.

8.5.1 Block connection
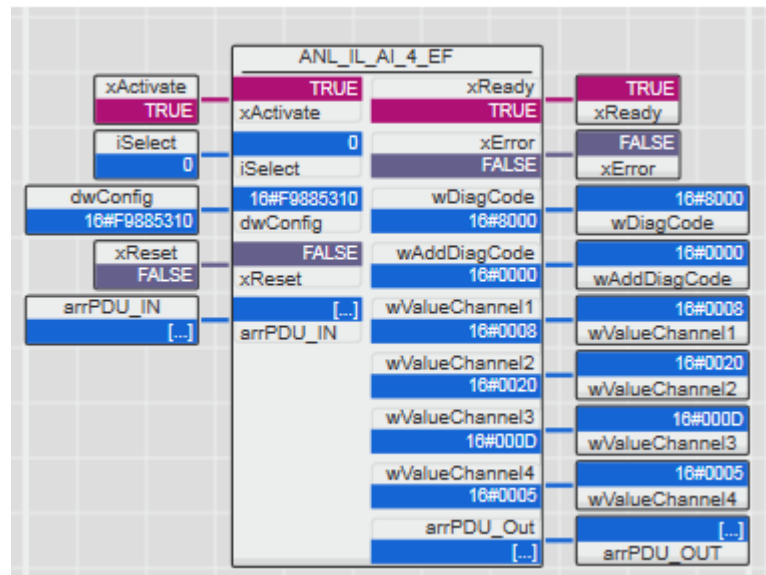
In the example the channels are configured as follows:

Channel	Range	Format	Filter
1	0-10 V	ST	Average 16
2	+/- 5 V	ST	No average
3	0-20 mA	IL	Average 4
4	+/- 20 mA	NORM	Average 32

The operating mode is set to "Read analog values" (iSelect = 0).

8.6 Appendix

8.6.1 Data types

```
TYPE
    AIEF_ARR_b_0_9 : ARRAY[0..9] OF BYTE;
END_TYPE
```

9 ANL_IL_AI_4_UI

The ANL_IL_AI_4_UI_Com function block is recommended for using the modules IB IL AI 4 /I-PAC (2700458) and IB IL AI 4 /U-PAC (2700459).

For easier handling, the ANL_IL_AI_4_UI_Para function block is available to parameterize the module.

The ANL_IL_AI_4_UI_Diag function block is indicating the standardized value and the diagnostic information in the event of an error.

9.1 Function blocks

The ANL_IL_AI_4_UI_Com block is used to parameterize modules IB IL AI 4 /I-PAC (2700458) and IB IL AI 4 /U-PAC (2700459). In addition, the block outputs the four inputs standardized in a structure. The modules can be parameterized by using a structure or with the help of the ANL_IL_AI_4_UI_Para function block.

The ANL_IL_AI_4_UI_Para function block can be used to select the measuring filter and measuring range. The measuring value is standardized according to the values set at rHighLimit and rLowLimit, and output as a real value.

The value is output in the udtStatus structure of the ANL_IL_AI_4_UI_Com function block or by using the ANL_IL_AI_4_UI_Diag function block at the rCurentValue output. In this block, the structure is split and the standardized real value is output. In addition, information about the validity of the value as well as an error description for invalid values can be found here.

If no parameters are set for iMode, iFilter, rHighLimit and/or rLowLimit, the default values of the module are accepted. All other input parameters require the use of values for operation.

The standard error codes and diagnostic messages are implemented. They are indicated via wDiagCode.

After block activation, the firmware version and the module number are indicated at strFirmware output additionally.

Changes to the parameters in the block are only implemented after a positive edge at the xActivate input.

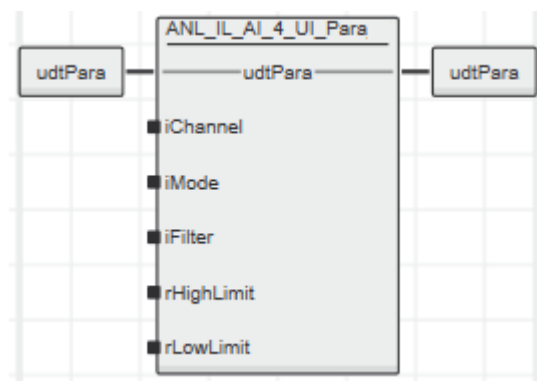An error state can be acknowledged with a positive edge at the xActivate input.

9.2 ANL_IL_AI_4_UI_Para

The IL_AI_4_UI_Para block is used to parameterize modules IB IL AI 4 /I-PAC (2700458) and IB IL AI 4 /U-PAC (2700459).

9.2.1 Change notes

Version	Change note
1	Converted from PC Worx 6

9.2.2 Block call

9.2.3 Input parameters

Name	Type	Description
iChannel	INT	Parameterizes the specified channel: 1 = Channel 1 is being parameterized. 2 = Channel 2 is being parameterized. 3 = Channel 3 is being parameterized. 4 = Channel 4 is being parameterized.
iMode	INT	Selects the input range: 0 = 0 V….10 V, 0 mA …. 20 mA (default) 1 = +-10 V, 4 mA …. 20 mA
iFilter	INT	Selects the filter: 0 = 16 times the average value (default) 1 = Filter is disabled 2 = 4 times the average value 3 = 32 times the average value
rHighLimit	REAL	Setting the upper standardized value.
rLowLimit	REAL	Setting the lower standardized value.

9.2.4 Inout parameters

Name	Type	Description
udtPara	ANL_UDT_Para_1_5	This structure acquires the parameters. CH1, CH2, CH3 and CH4 should be used for CHX. **udtPara.CHXiChannel:** CH1 = 1, CH2 = 2, CH3 = 3, CH4 = 4 CHX = X: Parameters are accepted CHX <> X: Parameters are not accepted **udtPara.CHXiMode:** Selects the input range: 0 = 0 V…. 10 V, 0 mA …. 20 mA (default) 1 = +-10 V, 4 mA…. 20 mA **udtPara.CHXiFilter:** Selects the filter 0 = 16 times the average value (default) 1 = Filter is disabled 2 = 4 times the average value 3 = 32 times the average value **udtPara.CHXrHighLimit:** Setting the upper standardized value. **udtPara.CHXrLowLimit:** Setting the lower standardized value.

9.3 ANL_IL_AI_4_UI_Com

The function block can be operated in various operating modes. The operating modes are set via iModul, iMode, iFilter and bSelect. However, bSelect only selects the channels for which the parameterization is to be applied.

9.3.1 Function block call
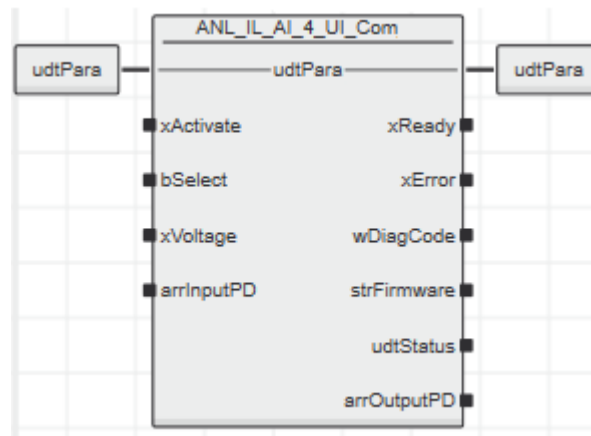
9.3.2 Input parameters

Name	Type	Description				
xActivate	BOOL	Rising edge: Activates the function block. FALSE: Deactivates the function block.				
bSelect	BYTE	Selects the inputs to be parameterized:				
		bSelect	Channel 4	Channel 3	Channel 2	Channel 1
		00				
		01				x
		02			x	
		03			x	x
		04		x		
		05		x		x
		06		x	x	
		07		x	x	x
		08	x			
		09	x			x
		0A	x		x	
		0B	x		x	x
		0C	x	x		
		0D	x	x		x
		0E	x	x	x	
		0F	x	x	x	x
xVoltage	BOOL	Selects the connected module TRUE: IB IL AI 4 /U-PAC (2700459) FALSE: IB IL AI 4 /I-PAC (2700458)				
arrInputPD	ANL_ARR_W_1_4	64-bit input of the module				

9.3.3 Output parameters

Name	Type	Description
xReady	BOOL	FALSE: The function block is executing services. TRUE: The function block is ready to execute services.
xError	BOOL	TRUE: An error has occurred. For details refer to wDiagCode and wAddDiagCode.
wDiagCode	WORD	Diagnosis code. Refer to diagnostic table.
strFirmware	STRING	Firmware version and module number
udtStatus	ANL_UDT_Status_1_5	This structure outputs the current module information. CH1, CH2, CH3 and CH4 should be used for CHX **udtStatus.CHXDataValid:** TRUE = The output value is valid. FALSE = The output value is invalid. **udtStatus.CHXDiagCode:** Provides information about the error type in the event of an error. **udtStatus.CHXCurentValue:** Provides the standardized analog value as a real value
arrOutputPD	ANL_ARR_W_1_4	64-bit output of the module

9.3.4 Inout parameters

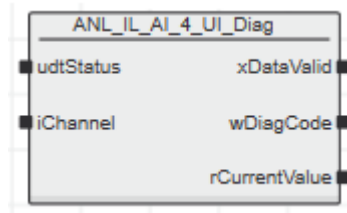| Name	Type	Description
udtPara	ANL_UDT_Para_1_5	This structure acquires the parameters. CH1, CH2, CH3 and CH4 should be used for CHX **udtPara.CHXiChannel:** CH1 = 1, CH2 = 2, CH3 = 3, CH4 = 4 CHX = X: Parameters are accepted CHX <> X: Parameters are not accepted **udtPara.CHXiMode:** Selects the input range: 0 = 0 V…. 10 V, 0 mA …. 20 mA (default) 1 = +-10 V, 4 mA…. 20 mA **udtPara.CHXiFilter:** Selects the filter 0 = 16 times the average value (default) 1 = Filter is disabled 2 = 4 times the average value 3 = 32 times the average value **udtPara.CHXrHighLimit:** Setting the upper standardized value. **udtPara.CHXrLowLimit:** Setting the lower standardized value.

9.3.5 Diagnosis

DiagCode	Description
16#0000	Function block is deactivated
16#8000	Function block is in regular operation
16#8001	Measuring range exceeded (overrange)
16#8002	Open circuit
16#8004	Measured value invalid/no valid measured value available
16#8010	Configuration invalid
16#8020	Supply voltage faulty
16#8040	Device faulty
16#8080	Below measuring range (underrange)
16#C100	Incorrect parameters
16#C101	Invalid parameter for iMode
16#C102	Invalid parameter for iFilter
16#C103	Invalid parameter for bSelect
16#C104	Invalid limit values at rHighLimit and rLowLimit

9.4 ANL_IL_AI_4_UI_Diag

The IL_AI_4_UI_Diag block is used for diagnostics.

9.4.1 Function block call

9.4.2 Input parameters

Name	Type	Description
iChannel	INT	Parameterizes the specified channel: 1 = Channel 1 is being parameterized. 2 = Channel 2 is being parameterized. 3 = Channel 3 is being parameterized. 4 = Channel 4 is being parameterized.
udtStatus	ANL_UDT_Status_1_5	This structure outputs the current module information. CH1, CH2, CH3 and CH4 should be used for CHX **udtStatus.CHXDataValid** TRUE = The output value is valid. FALSE = The output value is invalid. **udtStatus.CHXDiagCode** Provides information about the error type in the event of an error. **udtStatus.CHXCurrentValue** Provides the standardized analog value as a real value.

9.4.3 Output parameters

Name	Type	Description
xDataValid	BOOL	Indicates whether it was possible to calculate the specified value correctly TRUE = The analog value is valid. FALSE = The analog value is invalid.
wDiagCode	WORD	Diagnosis code. Refer to diagnostic table.
rCurrentValue	REAL	Standardized analog value

9.5 Startup instructions

For the startup instruction of the ANL_AI_4_UI function blocks please find the following examples:

- AnalogTechnology_10_EXA_ANL_IL_AI_4_UI_U.pcwex
- AnalogTechnology_10_EXA_ANL_IL_AI_4_UI_I.pcwex

These examples are located in the "Examples" folder of the unzipped msi file of the library.
They describe the use of the IB IL AI 4 /I-PAC (2700458) and IB IL AI 4 /U-PAC (2700459) modules in combination with the ANL_AI_4_UI function blocks.
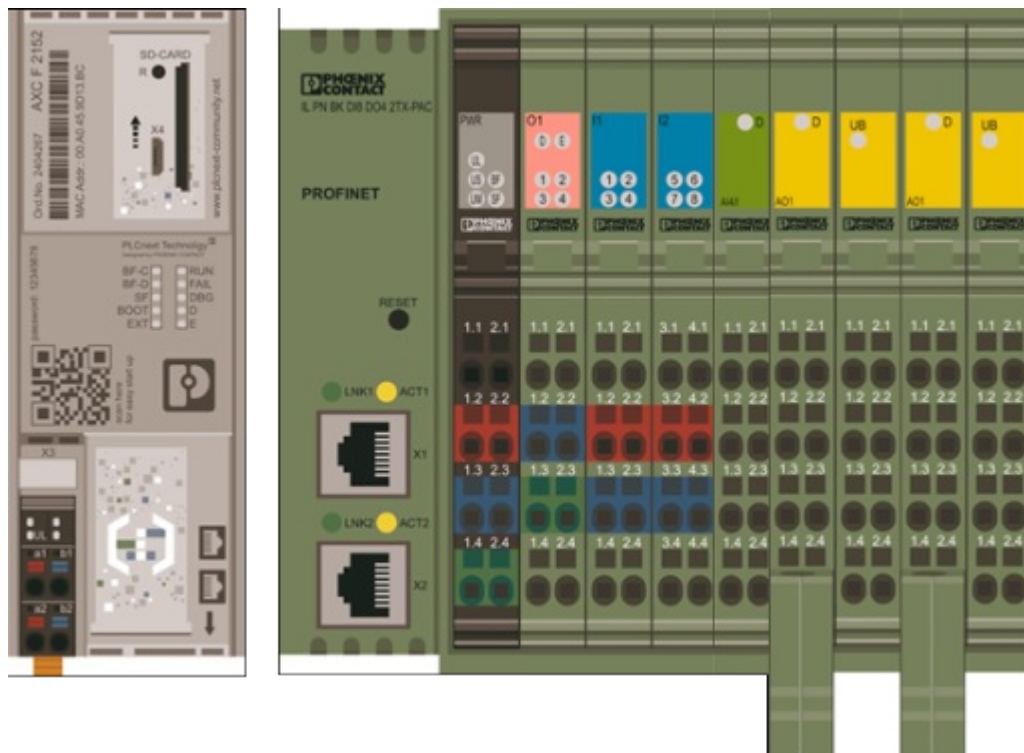
9.5.1 Startup example in voltage mode

The AnalogTechnology_10_EXA_ANL_IL_AI_4_UI_U.pcwex example shows the startup of the devices under voltage.

9.5.1.1 Bus structure

For this example, the following hardware is used:

- AXC F 2152 (2404267)
- IL PN BK DI8 DO4 2TX-PAC (2703994)
- IB IL AI 4 /U-PAC (2700459)
- IB IL AO 1/SF (2726298)
- IB IL AO 1/SF (2726298)

9.5.1.2 Block connection

For the power supply of channel 1-4, four block instances of the ANL_AO_1_SF function block are required and parameterized accordingly.

The ANL_IL_AI_4_UI_Para function blocks is used for writing the parameters.
In order to parameterize all four inputs, the ANL_IL_AI_4_UI_Para function block is called four times. In this example all channels are parameterized with 0-10 V.
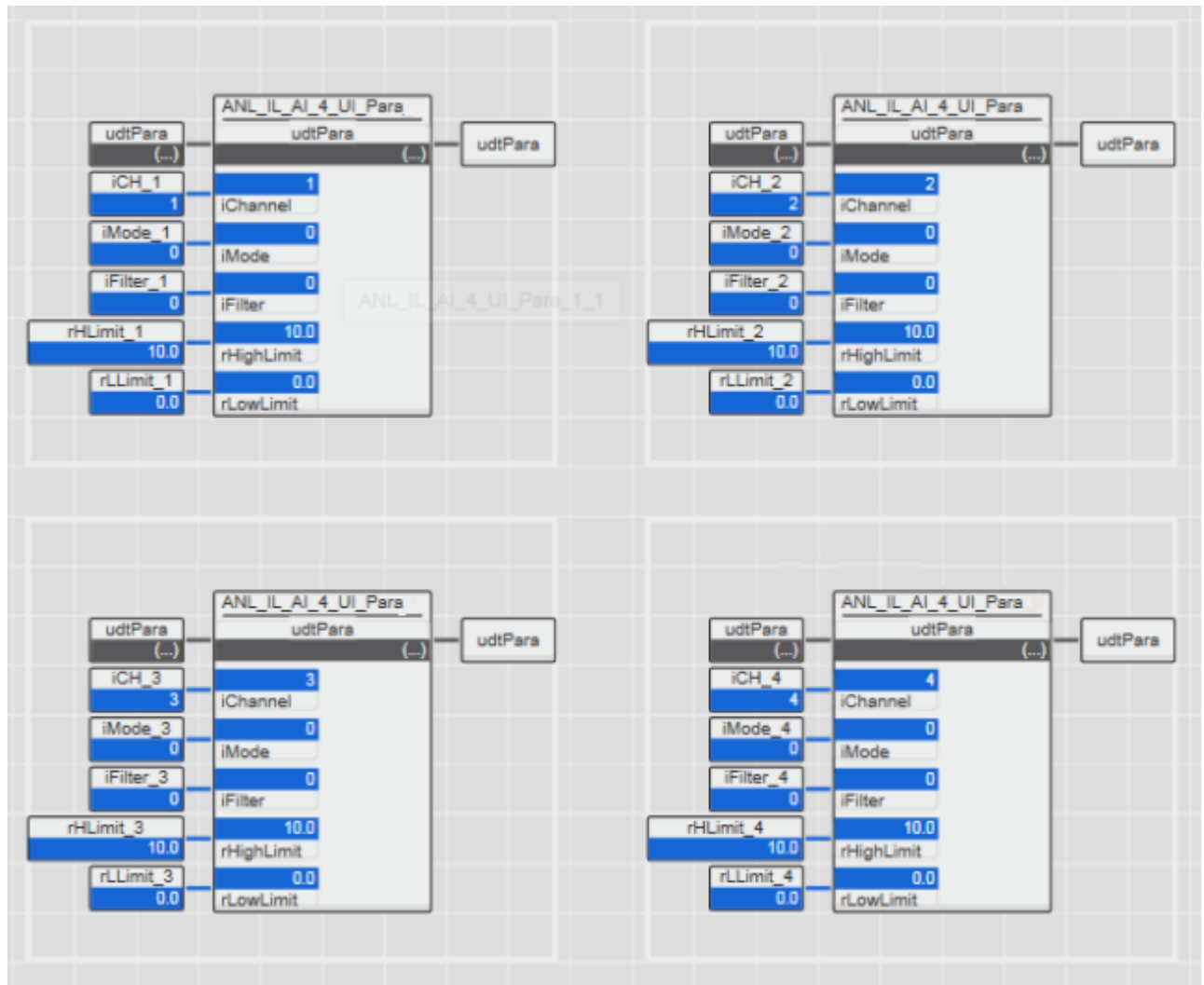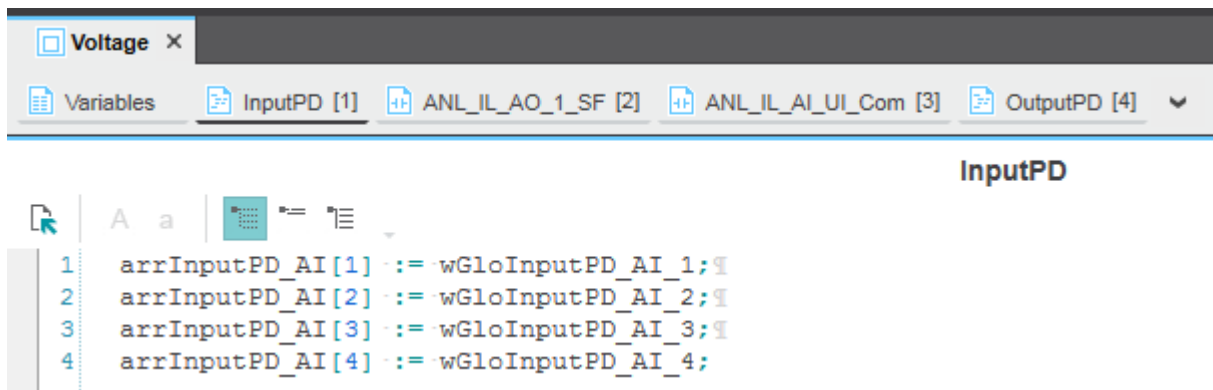

Afterwards the ANL_IL_AI_4_UI_Com block is called.
To parameterize all the inputs, value Byte#16#0F will be transferred to bSelect input. As the module used is an IB IL AI 4 /U-PAC (2700459) module, the xVoltage variable is set to TRUE. If the xActivate variable is now set to TRUE, the firmware version and the module number is indicated at the strFirmware output. If connection has been performed correctly and the parameters have been set correctly, signal 16#8000 appears at the wDiagCode output indicating an error-free state.

	ANL_IL_AI_4_UI_Com	
udtPara (...)	udtPara (...)	udtPara (...)
xActivate TRUE	TRUE xActivate	xReady TRUE	TRUE xReady
bSelect 16#0F	16#0F bSelect	xError FALSE	FALSE xError
xVoltage TRUE	TRUE xVoltage	wDiagCode 16#8000	16#8000 wDiagCode
arrInputPD_AI [...]	[...] arrInputPD	strFirmware '1.02 6'	'1.02 6' strFirmware
		udtStatus (...)	(...) udtStatus
		arrOutputPD [...]	[...] arrOutputPD_AI

To write the process data and read the input values, the variables at the arrInputPD and arrOutputPD are connected to the process data words of the module. Please note, since the process data cannot be linked to an array of word, it must be mapped using a global variable of type WORD.

Input process data:


```
1    arrInputPD_AI[1] := wGloInputPD_AI_1;
2    arrInputPD_AI[2] := wGloInputPD_AI_2;
3    arrInputPD_AI[3] := wGloInputPD_AI_3;
4    arrInputPD_AI[4] := wGloInputPD_AI_4;
```

Output process data:


```
1    wGloOutputPD_AI_1 := arrOutputPD_AI[1];
2    wGloOutputPD_AI_2 := arrOutputPD_AI[2];
3    wGloOutputPD_AI_3 := arrOutputPD_AI[3];
4    wGloOutputPD_AI_4 := arrOutputPD_AI[4];
```

For diagnostics, also the ANL_IL_AI_4_UI_Diag function block has to be called four times. Following that, all inputs and outputs are assigned variables. The udtPara inputs and outputs of all function blocks are connected to each other, as well as the udtStatus inputs and outputs. For the analog values to be indicated, the iChannel inputs of the ANL_IL_AI_4_UI_Diag function blocks are assigned the decimal values 1, 2, 3 and 4. The rCurrentValue variables of the ANL_IL_AI_4_UI_Diag function blocks output the standardized analog values finally.
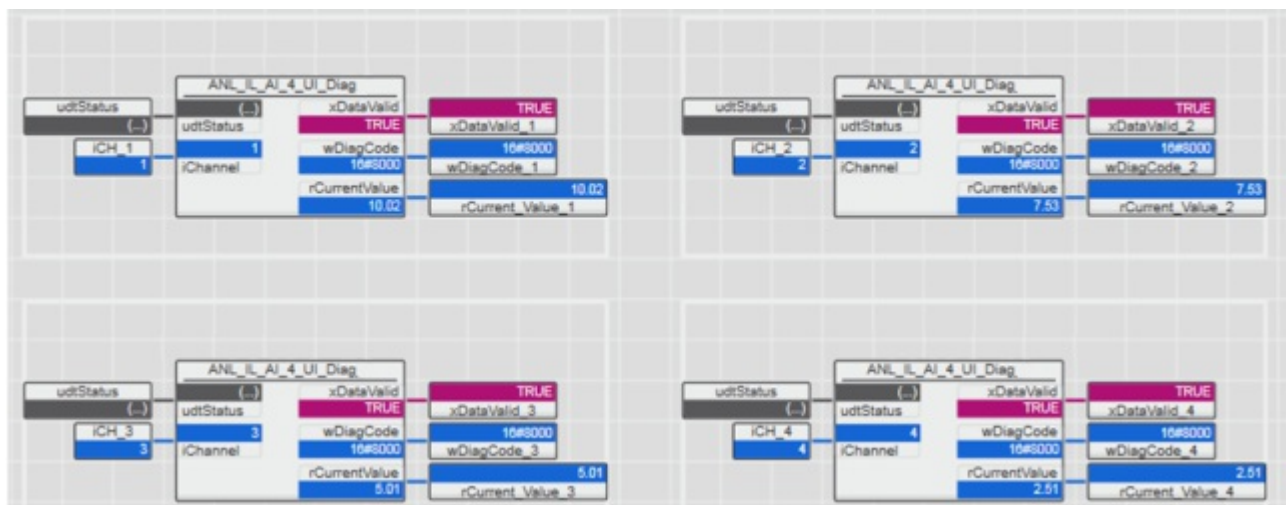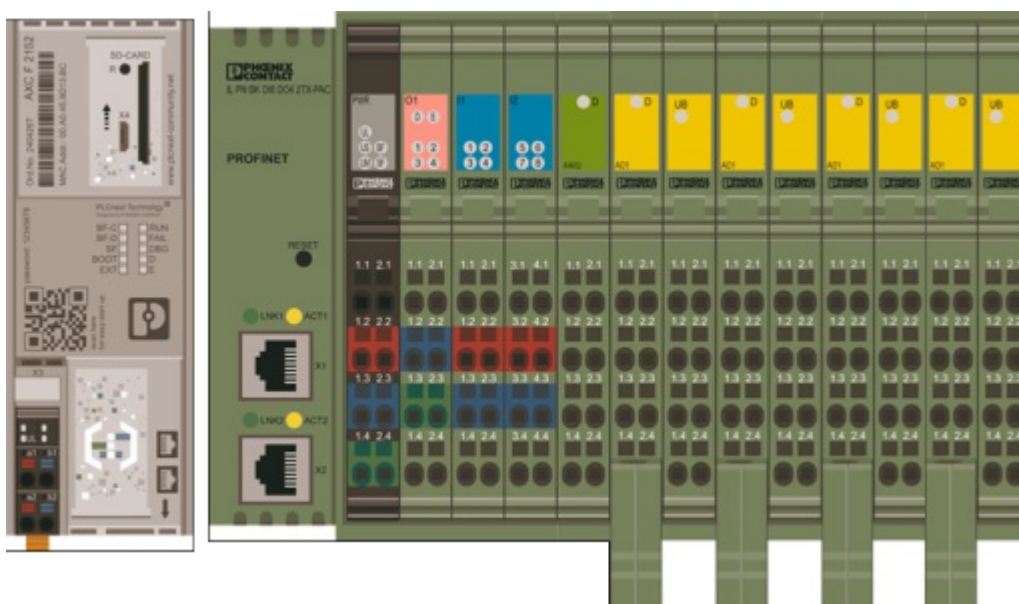
9.5.2 Startup example in current mode

The AnalogTechnology_10_EXA_ANL_IL_AI_4_UI_I.pcwex example shows the startup of the devices with power.

9.5.2.1 Bus structure

For this example, the following hardware is used:

- AXC F 2152 (2404267)
- IL PN BK DI8 DO4 2TX-PAC (2703994)
- IB IL AI 4 /I-PAC (2700458)
- IB IL AO 1/SF (2726298)
- IB IL AO 1/SF (2726298)
- IB IL AO 1/SF (2726298)
- IB IL AO 1/SF (2726298)

9.5.2.2 Block connection

For the power supply of channel 1-4, two block instances of the ANL_AO_1_SF function block are required and parameterized accordingly.
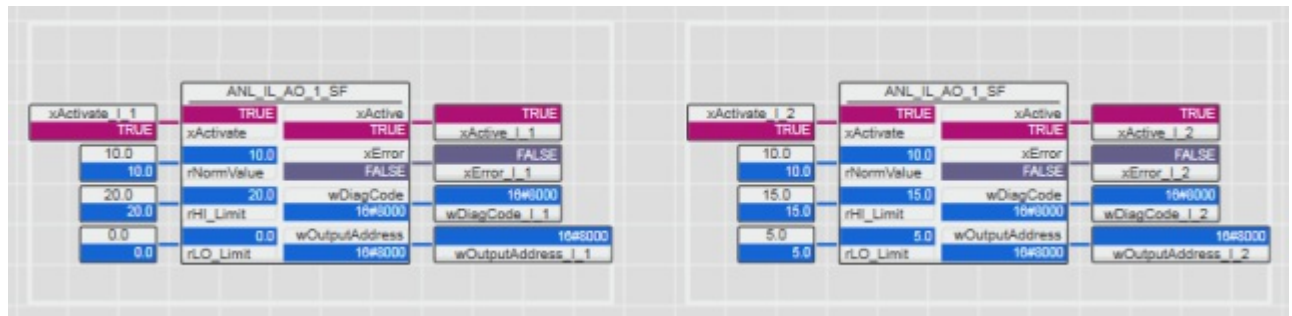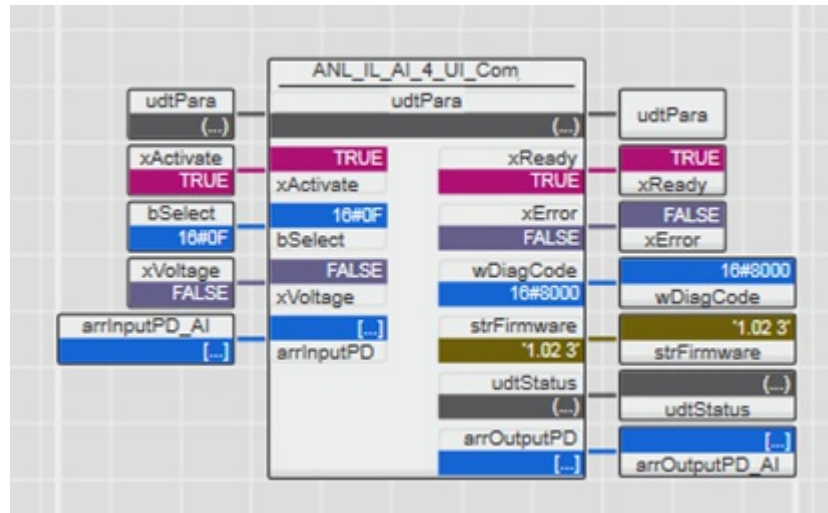

In order to parameterize all four inputs, the ANL_IL_AI_4_UI_Para function block is called four times. Here channel 1 and 3 are parameterized with 0-20 mA and channel 2-4 with 4-20 mA.

Afterwards the ANL_IL_AI_4_UI_Com block is called.
To parameterize all the inputs, value Byte#16#0F will be transferred to bSelect input. As the module used is an IB
IL AI 4 /I-PAC (2700458) module, the xVoltage variable is set to FALSE. If the xActivate variable is now set to
TRUE, the firmware version and the module number is indicated at the strFirmware output. If connection has been
performed correctly and the parameters have been set correctly, signal 16#8000 appears at the wDiagCode output
indicating an error-free state.

To write the process data and read the input values, the variables at the arrInputPD and arrOutputPD are connected
to the process data words of the module. Please note, since the process data cannot be linked to an array of word,
it must be mapped using a global variable of type WORD.

Input process data:


```
1    arrInputPD_AI[1] := wGloInputPD_AI_1;
2    arrInputPD_AI[2] := wGloInputPD_AI_2;
3    arrInputPD_AI[3] := wGloInputPD_AI_3;
4    arrInputPD_AI[4] := wGloInputPD_AI_4;
```

Output process data:


```
1    wGloOutputPD_AO_I1 := wOutputAddress_I_1;
2    wGloOutputPD_AO_I2 := wOutputAddress_I_2;
3
4    wGloOutputPD_AI_1 := arrOutputPD_AI[1];
5    wGloOutputPD_AI_2 := arrOutputPD_AI[2];
6    wGloOutputPD_AI_3 := arrOutputPD_AI[3];
7    wGloOutputPD_AI_4 := arrOutputPD_AI[4];
```

For diagnostics, also the ANL_IL_AI_4_UI_Diag function block has to be called four times. Following that, all inputs and outputs are assigned variables. The udtPara inputs and outputs of all function blocks are connected to each other, as well as the udtStatus inputs and outputs. For the analog values to be indicated, the iChannel inputs of the ANL_IL_AI_4_UI_Diag function blocks are assigned the decimal values 1, 2, 3 and 4. The rCurrentValue variables of the ANL_IL_AI_4_UI_Diag function blocks output the standardized analog values finally.
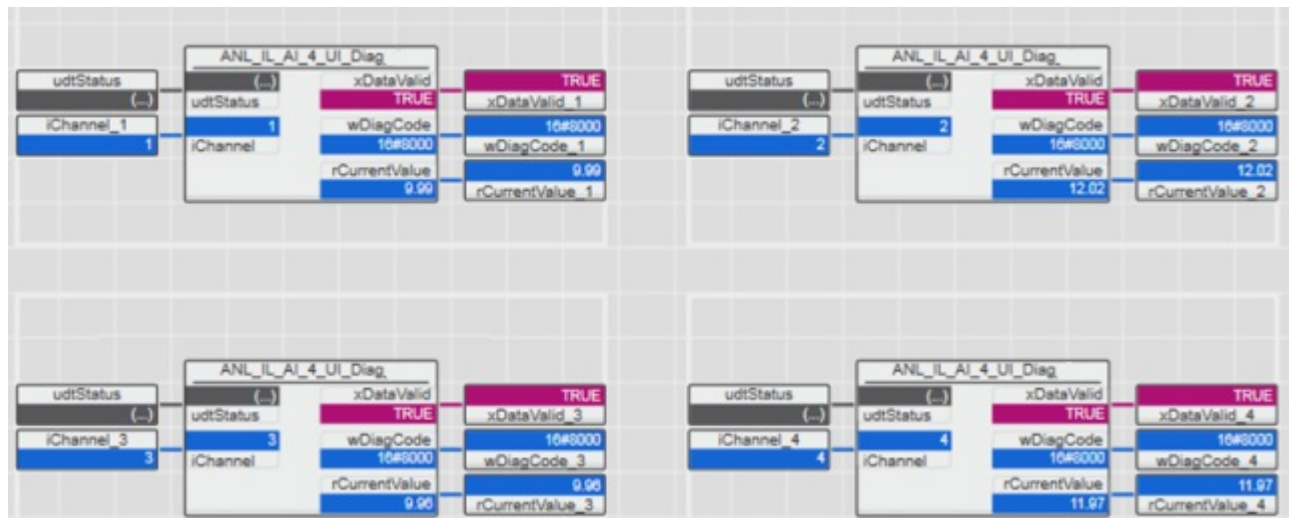
9.6 Appendix

9.6.1 Data types

```
TYPE

    ANL_udt_Status_1_5: STRUCT
        CH1xDataValid   : BOOL;
        CH1wDiagCode    : WORD;
        CH1rCurrentValue: REAL;
        CH1iChannel     : INT;
        CH2xDataValid   : BOOL;
        CH2wDiagCode    : WORD;
        CH2rCurrentValue: REAL;
        CH2iChannel     : INT;
        CH3xDataValid   : BOOL;
        CH3wDiagCode    : WORD;
        CH3rCurrentValue: REAL;
        CH3iChannel     : INT;
        CH4xDataValid   : BOOL;
        CH4wDiagCode    : WORD;
        CH4rCurrentValue: REAL;
        CH4iChannel     : INT;
    END_STRUCT

    ANL_udt_Para_1_5: STRUCT
        CH1iChannel     : INT;
        CH1iMode        : INT;
        CH1iFilter      : INT;
        CH1rHighLimit   : REAL;
        CH1rLowLimit    : REAL;
        CH1wDiagCode    : WORD;
        CH2iChannel     : INT;
        CH2iMode        : INT;
        CH2iFilter      : INT;
        CH2rHighLimit   : REAL;
        CH2rLowLimit    : REAL;
        CH2wDiagCode    : WORD;
        CH3iChannel     : INT;
        CH3iMode        : INT;
        CH3iFilter      : INT;
        CH3rHighLimit   : REAL;
        CH3rLowLimit    : REAL;
        CH3wDiagCode    : WORD;
        CH4iChannel     : INT;
        CH4iMode        : INT;
        CH4iFilter      : INT;
        CH4rHighLimit   : REAL;
        CH4rLowLimit    : REAL;
        CH4wDiagCode    : WORD;
    END_STRUCT

ANL_ARR_AI4UI_ChaDats : ARRAY[1..4] OF ANL_UDT_AI4UI_ChaDat;

    ANL_UDT_AI4UI_ChaDat :STRUCT    (* Internal data of one channel *)
        iChannel    : INT; (* Channel number *)
        iMode       : INT; (* Parameter *)
        iFilter     : INT; (* Parameter *)
        rHighLimit  : REAL; (* Parameter *)
        rLowLimit   : REAL; (* Parameter *)
        xConfigured : BOOL; (* TRUE = Channel is configured and configuration is valid *)
        xSelected   : BOOL; (* TRUE = Channel is selected by bSelect *)
```

```
        wDiagCode   : WORD; (* Diag code *)
        xDataValid  : BOOL; (* internal xDataValid *)
        wInput      : WORD; (* internal input process data *)
        wOut        : WORD; (* internal output process data *)
        rCurVal     : REAL; (* internal calculated current value *)
        xError      : BOOL; (* Error *)
    END_STRUCT
END_TYPE
```

10 ANL_IL_AI_8

The ANL_IL_AI_8 function block is used to parameterize and standardize measured values for the terminal types listed in the block data below.

A plausibility check is performed on the parameter data entered with this block.

Errors detected by the function block are displayed in the diagnostic register of the function block. These include device error messages and exceeding parameter data limit values. A block call for each Inline terminal is required.

10.1 Operating modes

10.1.1 Enabling

In order of using the block, integrate it into your application. A block call for each analog terminal is required. Activate it after the first PLC cycle. This is done by controlling the **xActivate** input variable from FALSE to TRUE. The channels are parameterized with a positive signal edge at **xActivate**. As soon as this has been completed successfully, this is indicated by **xReady** FALSE => TRUE.

Please note that at least one cycle is required with **xActivate** = FALSE for the internal variables to be initialized.

If the module is stopped (voltage failure, bus system stopped, etc.) it may be necessary to repeat parameterization. **xActivate** should be controlled as described above.

As soon as all the selected channels have been read in without any errors, this is indicated by the **xSelectValid** output parameter = TRUE. If a channel is read in with an error (e.g. limit value exceeded), the status of the **xSelectValid** output parameter = FALSE. The error cause can be determined using the **wDeviceDiagCode** block parameter.

10.1.2 Parameterizing analog channels

In order of parameterizing the channels the UDT_IL_AI8_Para parameter structure must be provided with the corresponding block parameters. The **xActivate** input parameter must have a positive edge. It can only be reparameterized with a positive edge. The end of parameterization is indicated by **xReady** (FALSE => TRUE). This procedure is also required to reset an error that has been removed. The function block performs a plausibility check on the specified parameter values. If invalid values have been specified for a parameter, the block generates an error message (**wDiagCode** and **wAddDiagCode**), which precisely identifies the faulty channel and parameter. The parameters are provided with the default values on initial instantiation of the block. The block parameter initialization values correspond to the default values of the analog terminals.
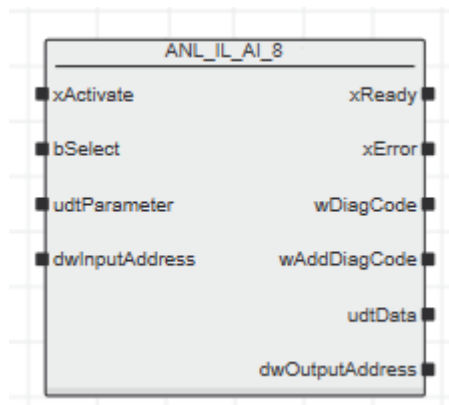
10.1.3 Outputting standardized measured values

In order to read measured values or error messages for channels 1 - 8 of an analog terminal, the selected input variable must be provided with a corresponding value.

The channels are read in order to channel 1 to 8. Therefore, the selected input parameter must be provided with a corresponding value. The activated bits of the selected parameter permit the channels to be read. Therefore, the latest significant bit (bit 0) corresponds to channel 1 the most significant bit (bit 7) corresponds to channel 8. Not selected channels are skipped. So used channels will be read faster.

In addition, standardization can be deactivated for certain channels with standardization block parameter = FALSE. This increases the speed, especially for slow CPUs (-> no floating point calculation). The measured values, validity indication, and channel-specific error messages for channels are indicated and output following activation of the block in the UTD_IL_AI8_Data structure of the Data output parameter. The values read by the terminal are analyzed by the function block and error messages are returned in the **Errorcode** variable. Measured values are standardized in the Value variables as a real value according to the limit values. The process data word (in IL format) for the channel is stored in the **Value_PD** variable. The **NDR** variable (FALSE => TRUE) indicates that a

new measured value has been read for a cycle. If an error is detected for a channel by the function block (e.g. limit value exceeded), the Error bit is controlled from FALSE to TRUE. The **Errorcode** contains the error code in IL format. Please note the last valid measured values are still displayed.

10.2 Function block call

10.3 Input parameters

Name	Type	Description
xActivate	BOOL	Block activation / deactivation. This parameter should be controlled by a value that represents the status of connected I/O devices (active/not active). This ensures that an incorrect diagnostic message is not generated by the block when I/O devices are disconnected. If block processing is activated (xActivate = TRUE) all the selected channels are read. The wDiagCode and wAddDiagCode diagnostic registers contain the status of the function and the terminal.
bSelect	BYTE	By activating the individual bits from 0 to 7 (correspond to channels 1 to 8), automatic reading in of the individual channels is activated.
udtParameter	ANL_ARR_IL_AI_8_Para	Array of Channel input parameter
dwInputAddress	DWORD	Process data input address for the terminal.

10.4 Output parameters

Name	Type	Description
xReady	BOOL	Block is ready, parameterization completed.
xError	BOOL	Error during initialization or operation.
wDiagCode	WORD	Diagnostic information.
wAddDiagCode	WORD	Additional diagnostic information.
xSelectValid	BOOL	Selected channels are ready.
udtData	ANL_ARR_IL_AI_8_Para	Measured value and error output.
dwOutputAddress	DWORD	Process data output address for the terminal.

10.5 Block parameters

10.5.1 Channel parameters

The table describes the UDT_IL_AI_8_Para data structure of the Parameter input parameter.

The channels can be configured with the Filter, Range, Standardization, HighLimit, and LowLimit parameters. To configure the filter, for example, use the following call at the Parameter input parameter: udtParameter[1].Filter. The number in the array corresponds to the relevant channel (1 to 8).

For the meaning of the configuration values, please refer to the data sheet.

Name	Type	Description
Filter	INT	Filter (value 0 to 3), user setting, see data sheet.
Range	INT	Measuring range (value 0 to 15), user setting, see data sheet1. Measuring range 8 to 10, 12 and 13 only allowed for the IB IL AI8/IS terminal.
Standardization	BOOL	TRUE: Standardization enabled, user setting.
HighLimit	REAL	Upper limit value, user setting.
LowLimit	REAL	Lower limit value, user setting.

An alternative to writing the parameter from the program to the array is to connect a preceding function block.

10.5.2 Measured values and channel diagnostics

The table describes the UTD_IL_AI_8_Data data structure of the data output parameter. The read measured value of a channel is represented as a real number in the Value parameter, the process data word is represented in Value_PD. The NDR bit can be used to determine whether the values have been updated. This bit is always only = TRUE for one cycle. In the event of a channel-specific error (Error), the cause must be removed. The error is then reset on the next read cycle.

For the **Errorcode**, please refer to the data sheet of the module.

Name	Type	Description
ValuePD	WORD	Measured value of a channel.
Value	REAL	Measured value of a channel.
Error	BOOL	Channel indicates an error.
DeviceDiagCode	WORD	Error code for the channel (IL format), see data sheet.
NDR	BOOL	TRUE = New measured value present.

An alternative to calling the measured value information for the program from the array is to connect to a subsequent function block.

10.6 Diagnosis

wDiagCode	wAddDiagCode	Description
16#0000		Block is not active.
16#8000		Block is active.
16#C010		Timeout during communication
	16#0001	Module not responding or indicates an error Range 0-6 at terminal IB IL AI8/IS
16#C014		Filter - range limit exceeded
	16#CCXX	CC = channel, XX = range value
16#C017		Range - limit exceeded
	16#CCXX	CC = channel, XX = range value
16#C018		LowLimit >= HighLimit
	16#00CC	CC = channel

If several errors occur simultaneously, only the first error is indicated initially. The other error messages are displayed once the first error has been removed and acknowledged. The error is reset by generating a positive edge at xActivate. This process is repeated until all the errors have been removed.

10.7 Appendix

10.7.1 Data types

```
TYPE
    (*UDT_IL_AI_8_Para*)
    ANL_UDT_AI_8_IN_V_1      : STRUCT
        Filter               : INT; (*Filter by mean-value generation*)
        Range                : INT; (*Current measuring range*)
        Standardization      : BOOL; (*TRUE: Standardization enabled*)
        LoLim                : REAL; (*Lower limit value*)
        HiLim                : REAL; (*Upper limit value*)
    END_STRUCT;

    (*UDT_IL_AI_8_Data*)
    ANL_UDT_AI_8_OUT_V_1     : STRUCT
        Value_PD             : WORD; (*Measured value as WORD*)
        Value                : REAL; (*Measured value as REAL*)
        Error                : BOOL; (*TRUE: Channel indicates an error*)
        ErrorCode            : WORD; (*Error code of the channel*)
        NDR                  : BOOL; (*TRUE: New value present*)
    END_STRUCT;

    (*Array of UDT_IL_AI_8_Para*)
    ANL_ARR_IL_AI_8_Para : ARRAY[1..8]      OF ANL_UDT_AI_8_IN_V_1;
    (*Array of UDT_IL_AI_8_Data*)
    ANL_ARR_IL_AI_8_Data : ARRAY[1..8]      OF ANL_UDT_AI_8_OUT_V_1;
    (*Array of Bool for selected channels*)
    ANL_ARR_IL_AI_8_SelectCH : ARRAY[1..8]  OF BOOL;
END_TYPE
```

10.7.2 Error codes of the module

Code(hex)	Error
16#8001	Overrange
16#8002	Open circuit
16#8004	Measured value invlaid/no valid measured value available
16#8010	Invalid configuration
16#8020	I/O supply voltage fault
16#8040	Module faulty
16#8001	Underrange

11 ANL_IL_AO_1_SF

The ANL_IL_AO_1_SF function block converts a standardized value at the analog output of the IB IL AO 1/SF (2726298) and IB IL AO 1/U/SF (2727776) modules.

The ANL_IL_AO_1_SF function block outputs a standardized analog value to the analog output of the IB IL AO 1/SF (2726298) or IB IL AO 1/U/SF (2727776) module.

Example:

Measuring range	Parameterized rLO_Limit	Parameterized rHI_Limit	Applied rNormValue value	Output at analog output
0-10V	2000.0	4000.0	2000.0	0.0
0-10V	2000.0	4000.0	3000.0	5.0
0-10V	2000.0	4000.0	4000.0	10.0

The function block cyclically converts the rNormValue values into analog values for the Phoenix Contact analog modules IB IL AO 1/SF (2726298) or IB IL AO 1/U/SF (2727776). Errors and error codes will be reported by time if an error occures. The IB IL AO 1/U/SF (2727776) and IB IL AO 1/SF (2726298) Inline modules provide one analog output with a voltage range from 0 V to 10 V. Additionally, the IB IL AO 1/SF (2726298) Inline module provides a current range from 0 mA to 20 mA or 4 mA to 20 mA. The different ranges can be tapped from different connecting points of the module.

11.1 Function block call
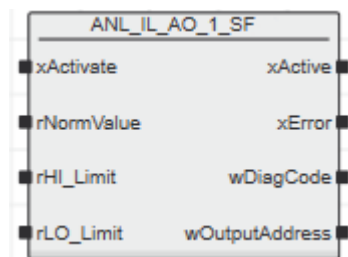
11.2 Input parameters

Name	Type	Description
xActivate	BOOL	Rising edge: Activates the function block. FALSE: Deactivates the function block.
rNormValue	REAL	Analog value to be output (value in the range between rLO_Limit and rHI_Limit)
rHI_Limit	REAL	High limit of the normed range
rLO_Limit	REAL	Low limit of the normed range

11.3 Output parameters

Name	Type	Description
xActive	BOOL	FALSE: Function block is not active. TRUE: Function block is active.
xError	BOOL	TRUE: An error has occurred. For details refer to wDiagCode and wAddDiagCode.
wDiagCode	WORD	Diagnosis code. Refer to diagnostic table.
wOutputAddress	WORD	Output variable of analog channel

11.4 Diagnosis

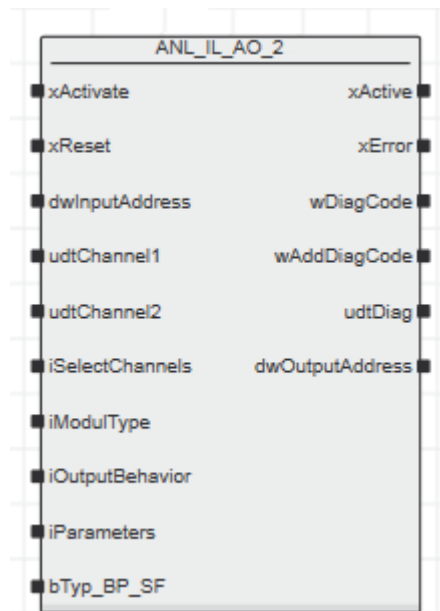| DiagCode	Description
16#0000	Function block is deactivated
16#8000	Function block is in regular operation
C030hex	Error from the function block: The limits are not correct. rHILimit <= rLOLimit
C040hex	Error from the function block: High limit overranged rIN_AO > rHILimit
C050hex	Error from the function block Low limit underranged rIN_AO < rLO_Limit

12 ANL_IL_AO_2

The ANL_IL_AO_2 function block serves for parameterization and operation of the Inline analog output modules IB IL AO 2 /U/BP-PAC (2861467) and IB IL AO 2 /SF-PAC (2863083).

The IB IL AO 2 /U/BP-PAC (2861467) and IB IL AO 2 /SF-PAC (2863083) Inline analog output modules can be operated together with the ANL_IL_AO_2 function block.

The function block is responsible for the following tasks:

- Parameterization of the the analog module,
- standardization and output of the analog values for channel 1 and/or channel 2 and
- the diagnostic function by outputting diagnostic codes.

12.1 Function block call

```
             ANL_IL_AO_2
─■ xActivate                      xActive ■─
─■ xReset                          xError ■─
─■ dwInputAddress              wDiagCode ■─
─■ udtChannel1              wAddDiagCode ■─
─■ udtChannel2                    udtDiag ■─
─■ iSelectChannels        dwOutputAddress ■─
─■ iModulType
─■ iOutputBehavior
─■ iParameters
─■ bTyp_BP_SF
```

12.2 Input parameters

The ANL_IL_AO_Channel function block is to be used for connection or transmission of the standardized values and their upper and lower limits. With the help of the block, three values (one for each channel) will be made available by structure to the ANL_IL_AO_2 function block at the udtChannelX input parameter.

Name	Type	Description
xActivate	BOOL	Rising edge: Activates the function block. FALSE: Deactivates the function block.
xReset	BOOL	Rising edge: Resets the function block.
dwInputAddress	DWORD	Channel 1+2: Input process data double word
udtChannel1	ST_Value	Structure with the analog values, upper and lower limits to be outputted (for channel 1)
udtChannel2	ST_Value	Structure with the analog values, upper and lower limits to be outputted (for channel 2)
iSelectChannels	INT	Channel selection: 1 = channel 1 2 = channel 2 3 = channel 1 and channel 2
iModulType	INT	Module type: 1 = IB IL AO 2/U/BP (unipolar) 2 = IB IL AO 2/U/BP (bipolar) 3 = IB IL AO 2/SF
iOutputBehavior	INT	Behavior of the outputs in the event of an error: 0 = Outputs hold the last value (Hold) 1 = Outputs are set to 0 (Reset)
iParameters	INT	Parameterization of the terminal: 0 = volatile parameterization 1 = non-volatile parameterization
bTyp_BP_SF	BOOL	Parameterization/selection of hardware used FALSE = IB IL AO 2/U/BP TRUE = IB IL 2/SF

12.3 Output parameters

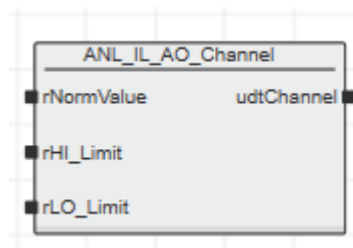| Name	Type	Description
xActive	BOOL	FALSE: Function block is not active. TRUE: Function block is active.
xError	BOOL	TRUE: An error has occurred. For details refer to wDiagCode and wAddDiagCode.
wDiagCode	WORD	Diagnosis code. Refer to diagnostic table.
wAddDiagCode	WORD	Additional diagnosis code. Refer to diagnostic table.
udtDiag	ANL_UDT_AO_DIAG	Additional diag information: iState = State of the internal state machine
dwOutputAddress	DWORD	Channel 1 + 2: Output process data double word

12.4 Diagnosis

wDiagCode	wAddDiagCode	Description
16#0000	16#0000	Function block is deactivated.
16#8000	16#0000	Function block is in regular operation.
16#C200		Error during activation of the block
	16#0001	Channel 1: Incorrect parameterization of the upper limit <= lower limit (rHI_Limit_CH1 <= rLO_Limit_CH1)
	16#0002	Channel 2: Incorrect parameterization of the upper limit <= lower limit (rHI_Limit_CH2 <= rLO_Limit_CH2)
	16#0003	Incorrect parameterization channel selection (iSelectChannels)
	16#0004	Incorrect parameterization: Module type (iModulType)
	16#0005	Incorrect parameterization: Volatile or non-volatile parameterization (iParameters)
	16#0006	Incorrect parameterization: Output behavior in the case of error (iOutputBehavior)
	16#0007	Analog module does not acknowledges parameterization within 5 seconds -> Timeout

Possible errors:
Incorrect module type was parameterized
Incorrect process data assignment |
	16#0008	The parameters iModulType and bTyp_BP_SF do not correspond
16#C300		Error in run
	16#0001	Channel 1: Input value < lower limit
	16#0002	Channel 1: Input value > upper limit
	16#0003	Channel 2: Input value < lower limit
	16#0004	Channel 2: Input value > upper limit
16#FFFF		Error in hardware
	16#8010	Invalid configuration or wrong wire bridges
	16#8020	Error in the I/O voltage supply.
	16#8040	Module is defect.

12.5 ANL_IL_AO_Channel

The function block ANL_IL_AO_Channel serves to determine the limit values. Corresponding data is made available to the function block ANL_IL_AO_2 as a structure.

12.5.1 Function block call

```
         ANL_IL_AO_Channel
 ■ rNormValue              udtChannel ■

 ■ rHI_Limit

 ■ rLO_Limit
```

12.5.2 Input parameters

Name	Type	Description
wNormValue	WORD	Analog value to be outputted (value in a range rLO_Limit and rHI_Limit)
rHI_Limit	REAL	Upper limit value
rLO_Limit	REAL	Lower limit value

12.5.3 Output parameters

Name	Type	Description
udtChannel	ST_Value	Structure with the values analog value to be outputted, upper and lower limits

12.6 Startup instructions

For the startup instruction of the ANL_IL_AO_2 function block the following example is available:
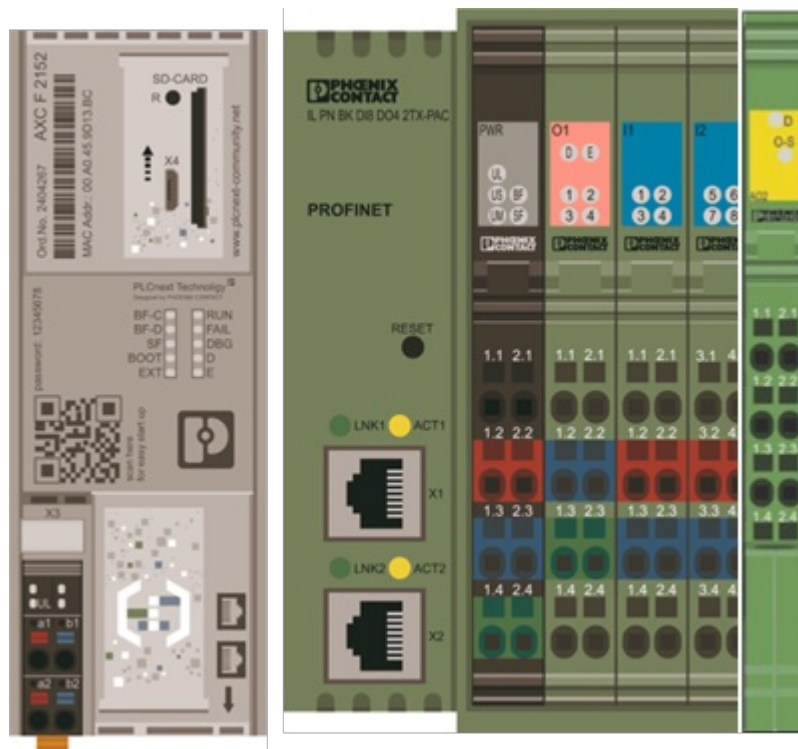
- AnalogTechnology_10_EXA_ANL_IL_AO_2.pcwex

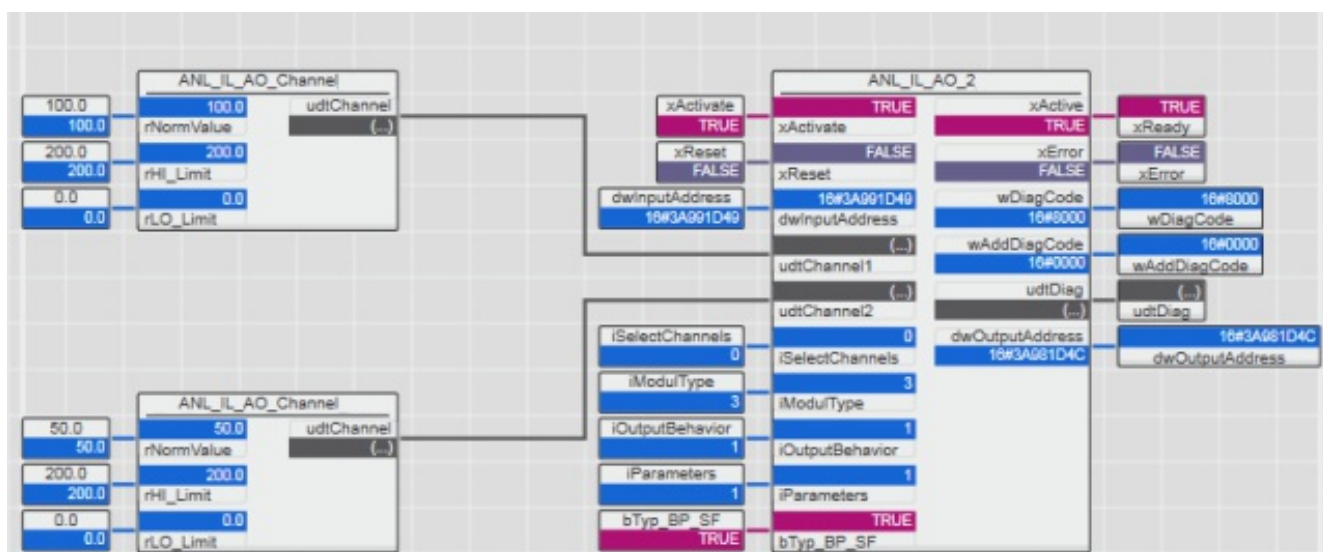The example is located in the "Examples" folder of the unzipped msi-file of the library.

12.6.1 Bus structure

For this example, the following hardware is used:

- AXC F 2152 (2404267)
- IL PN BK DI8 DO4 2TX-PAC (2703994)
- IB IL AO 2 /SF-PAC (2863083)

12.6.2 Operation of two channels

ANL_IL_AO_Channel
- 100.0 → rNormValue (100.0)
- 200.0 → rHI_Limit (200.0)
- 0.0 → rLO_Limit (0.0)
- udtChannel (...)

ANL_IL_AO_Channel
- 50.0 → rNormValue (50.0)
- 200.0 → rHI_Limit (200.0)
- 0.0 → rLO_Limit (0.0)
- udtChannel (...)

ANL_IL_AO_2
- xActivate TRUE → xActivate (TRUE)
- xReset FALSE → xReset (FALSE)
- dwInputAddress 16#3A991D49 → dwInputAddress (16#3A991D49)
- udtChannel1 (...)
- udtChannel2 (...)
- iSelectChannels 0 → iSelectChannels (0)
- iModulType 3 → iModulType (3)
- iOutputBehavior 1 → iOutputBehavior (1)
- iParameters 1 → iParameters (1)
- bTyp_BP_SF TRUE → bTyp_BP_SF (TRUE)
- xActive TRUE → xReady (TRUE)
- xError FALSE → xError (FALSE)
- wDiagCode 16#8000 → wDiagCode (16#8000)
- wAddDiagCode 16#0000 → wAddDiagCode (16#0000)
- udtDiag (...) → udtDiag (...)
- dwOutputAddress 16#3A981D4C → dwOutputAddress (16#3A981D4C)

12.6.3 Parameterization

The function block automatically carries out parameterization of the analog terminal after activation. For this, the corresponding parameters must be entered into the block. A changed parameterization will be either assumed by a positive edge on the input parameter xActivate or, in the case of an activated block, a positive edge on the input parameter xReset.

The following Inline analog output terminals can be operated with the blocks described:

Module types*	Data width	Polarity	Format	Additions
IB IL AO 2 /U/BP (2732732) IB IL AO 2 /U/BP-PAC (2861467)	12 bits	unipolar 0-10V	IL	1 or 2 channels
IB IL AO 2 /U/BP (2732732) IB IL AO 2 /U/BP-PAC (2861467)	13 bits	bipolar +/- 10 V	IL	1 or 2 channels
IB IL AO 1/SF (2726298) IB IL AO 1/SF-PAC (2861315)	16 bits	unipolar 0-10 V 0-20 mA 4-20 mA	IL	1 or 2 channels

*Parameterize the block with the corresponding value for the deployed Inline analog output terminals and operating mode.

Additional information on parameterization of the Inline analog output terminals can be found in the data sheets. The block only supports IL format.

12.6.4 Output of standardized analog values

The function block standardizes the input values within the stated limit values from the user (rLO_Limit and rHI_Limit) and outputs them over the Inline analog output terminal as a corresponding analog value.

12.6.5 Diagnostics/Error

In case of an error during the parameterization, the operation or a hardware error, the function block indicates a fault and the error bit xError will be set to 1. Here, the detected error code on the wDiagCode and wAddDiagCode parameters will be displayed. At the same time, the OutputAddress parameter (output double word) will be set to 16#00000000. In the event of an error and depending on parameterization (see data sheet on output behavior) the Inline analog output terminal outputs the last valid voltage or current. Values of the channels (parameterization HOLD) or both channels will be set to 0 (parameterization RESET).

The following diagnostics codes will be displayed during error-free operation:

- wDiagCode = 16#8000
- wAddDiagCode = 16#0000

12.7 Appendix

12.7.1 Data types

```
TYPE
    ST_Value : STRUCT
        rNormValue  : REAL; (* Input raw value *)
        rHI_Limit   : REAL; (* Upper limit *)
        rLO_Limit   : REAL; (* Lower limit *)
    END_STRUCT;

    ANL_UDT_AO_DIAG : STRUCT
        iState  : INT;
    END_STRUCT;
END_TYPE
```

13 ANL_IL_AO_2_UI

The function block is used for parameterizing and operating the IB IL AO 2/UI-PAC (2700775) Inline analog output terminal. The input values within the value range (rLoLim and rHiLim) that is set by the user, are standardized and are output as corresponding analog value by the Inline analog output terminal.

The following output ranges are supported:

- Unipolar: 0-10 V/ 0 - 20 mA/ 4 - 20 mA
- Bipolar: -10 V - +10 V/ -20 mA - +20 mA

The function block only supports the IB IL format.

The IB IL AO 2/UI-PAC (2700775) Inline analog output terminal can be operated with the ANL_IL_AO_2_UI function block.

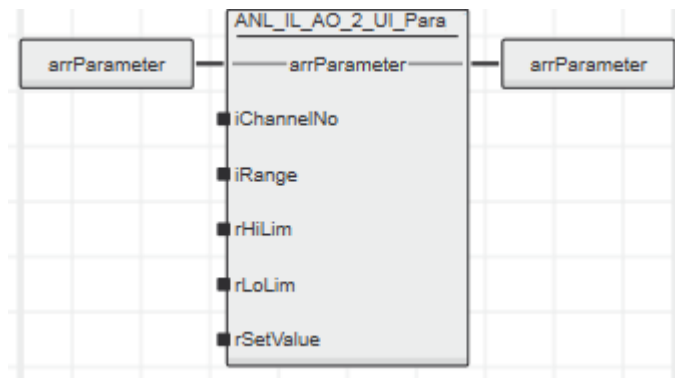The tasks of the function block are the following:

- Parameterization of the analog terminal,
- standardization and output of analog values for channel 1 and/or channel 2 and
- diagnostic function by the output of diagnosis codes.

For the setting or transmission of standardization values and their upper and lower limits, the ANL_IL_AO_UI_2_Para function block has to be used. Using the ANL_IL_AO_2_UI function block the values are provided at the input parameter arrParameter by structure. For differentiated channel diagnostics, the ANL_IL_AO_UI_2_Diag function block can be used. Using this function block the data of the arrDiag structure of ANL_IL_AO_2_UI function block can be read out and provided more easily.

13.1 ANL_IL_AO_2_UI_Para

The ANL_IL_AO_2_UI_Para function block defines the limit values and the output range. The corresponding data for the ANL_IL_AO_2_UI_Para function block is provided by structure. The illustration in the startup instructions shows the setting for both channels.

13.1.1 Function block call

13.1.2 Input parameters

Name	Type	Description
iCannelNo	INT	Channel selection • 1 Channel 1 • 2 Channel 2
iRange	INT	Output range • 0 0 V to 10 V • 1 -10 V to +10 V • 2 0 mA to 20 mA • 3 -20 mA to +20 mA • 4 4 mA to 20 mA
rHiLim	REAL	Upper limit value of the standardized range.
rLoLim	REAL	Lower limit value of the standardized range.
rSetValue	REAL	Analog value to be output (value in the range between /"rLoLim/" and /"rHiLim/").
arrParameter	ANL_ARR_Para_1_2	Structure with the values channel selection, output range, analog value to be output, upper and lower limit.

13.1.3 Inout parameters

Name	Type	Description
arrParameter	ANL_ARR_Para_1_2	Structure with the values channel selection, output range, analog value to be output, upper and lower limit.

13.2 ANL_IL_AO_2_UI

The function block is used for parameterizing and operating the IB IL AO 2/UI-PAC (2700775) Inline analog output terminal. The input values within the value range (rLoLim and rHiLim) that is set by the user, are standardized and are output as corresponding analog value by the Inline analog output terminal.

The following output ranges are supported:

- Unipolar: 0-10 V/ 0 - 20 mA/ 4 - 20 mA
- Bipolar: -10 V - +10 V/ -20 mA - +20 mA

The function block only supports the IB IL format.

13.2.1 Function block call

13.2.2 Input parameters

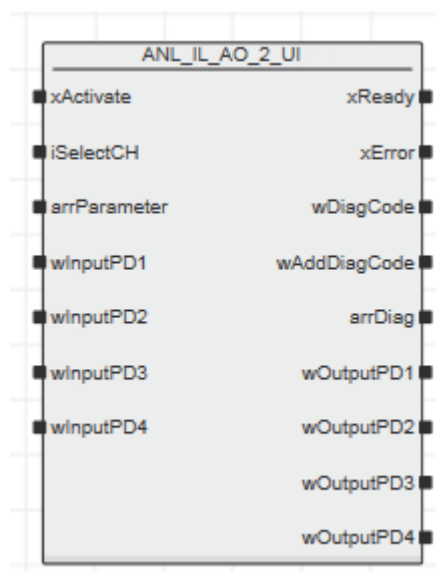| Name	Type	Description
xActivate	BOOL	Rising edge: Activates the function block. FALSE: Deactivates the function block.
iSelectCH	INT	Channel selection • 1 Channel 1 • 2 Channel 2 • 3 Channel 1 and 2
arrParameter	ANL_ARR_Para_1_2	Structure with the values channel selection, output range, analog value to be output, upper and lower limit.
wInputPD1	WORD	Input process data
wInputPD2	WORD	Input process data
wInputPD3	WORD	Input process data
wInputPD4	WORD	Input process data

13.2.3 Output parameters

Name	Type	Description
xReady	BOOL	The block is ready for operation and working without errors.
xError	BOOL	TRUE: An error has occurred. For details refer to wDiagCode and wAddDiagCode.
wDiagCode	WORD	Diagnosis code. Refer to diagnostic table.
wAddDiagCode	WORD	Additional diagnosis code. Refer to diagnostic table.
arrDiagCode	ANL_ARR_1_2	Structure with error message of the analog terminal during the operation for channel 1 and channel 2.
wOutputPD1	WORD	Output process data
wOutputPD2	WORD	Output process data
wOutputPD3	WORD	Output process data
wOutputPD4	WORD	Output process data

13.2.4 Diagnosis

wDiagCode	wAddDiagCode	Description
16#0000	16#0000	Function block is deactivated.
16#8000		Function block is in regular operation
16#C100		Configuration errors
	16#0001	Incorrect parameterization of the channel selection iSelectCH.
	16#0002	Incorrect parameterization of the output range.
	16#0003	Incorrect parameterization of the limit values (HiLim <= LoLim)
	16#0004	Timeout. Possible errors: An incorrect module type has been parameterized or the process data assignment is incorrect.
	16#0005	Incorrect parameterization of the channel selection iChannelNo.
	16#0006	Incorrect parameterization of the limit values and the analog value to be output.
16#C200		Errors in run.
	16#0001	Analog input value > upper limit value.
	16#0002	Analog input value < lower limit value.

13.3 ANL_IL_AO_2_UI_Diag

The function block ANL_IL_AO_2_UI_Diag is used for the simplified readout of data from the arrDiag structure of the ANL_IL_AO_2_UI function block. The output structure is assigned to the function block ANL_IL_AO_2_UI_Diag and dissolved in single output variables.

13.3.1 Function block call
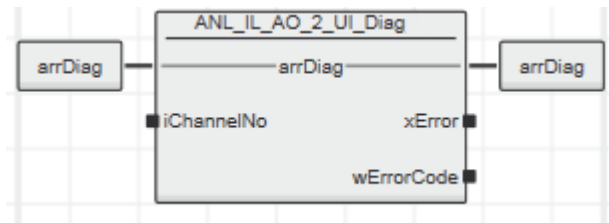
13.3.2 Input parameters

Name	Type	Description
iCannelNo	INT	Channel selection • 1 Channel 1 • 2 Channel 2

13.3.3 Output parameters

Name	Type	Description
xError	BOOL	TRUE: An error has occurred. For details refer to wDiagCode and wAddDiagCode.
wErrorCode	WORD	Error message of the analog terminal during the operation for channel 1 and channel 2: • 8001hex: Measuring range exceeded (Overrange). • 8002hex: Broken wire (only in operating mode 4-20mA). • 8003hex: Short circuit/overload of the output. • 8010hex: Invalid configuration. • 8020hex: Supply voltage incorrect. • 8040hex: Device is defective. • 8080hex: Below measuring range(Underrange).

13.3.4 Inout parameters

Name	Type	Description
arrDiag	ANL_ARR_Diag_1_2	Structure with the values channel selection, error message of the analog terminal for channel 1 and channel 2.

13.4 Startup instructions

For the startup instruction of the ANL_IL_AO_2_UI function block the following example is available:
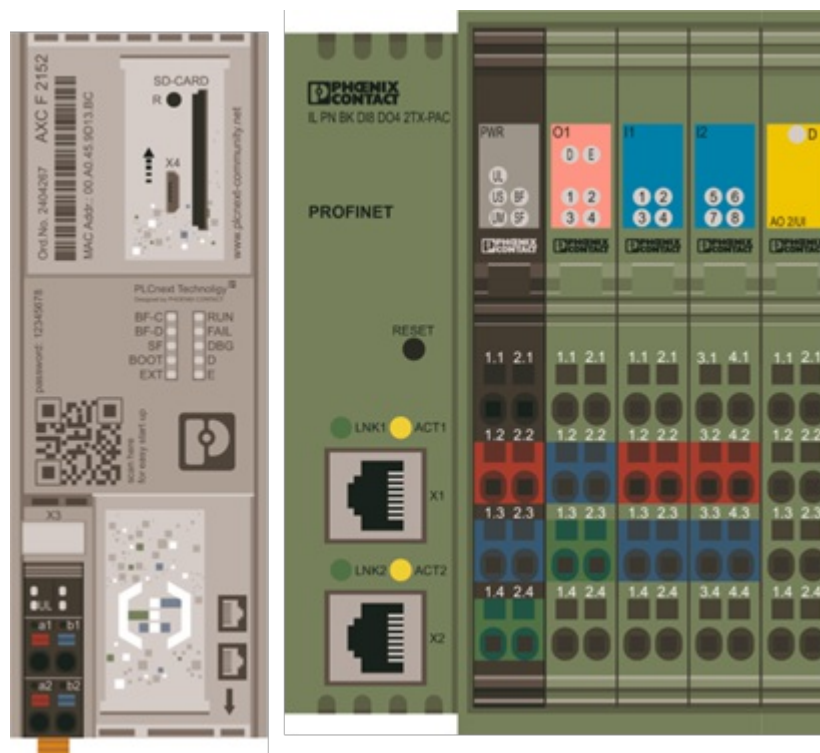
- AnalogTechnology_10_EXA_ANL_IL_AO_2_UI

The example is located in the "Examples" folder of the unzipped msi-file of the library.
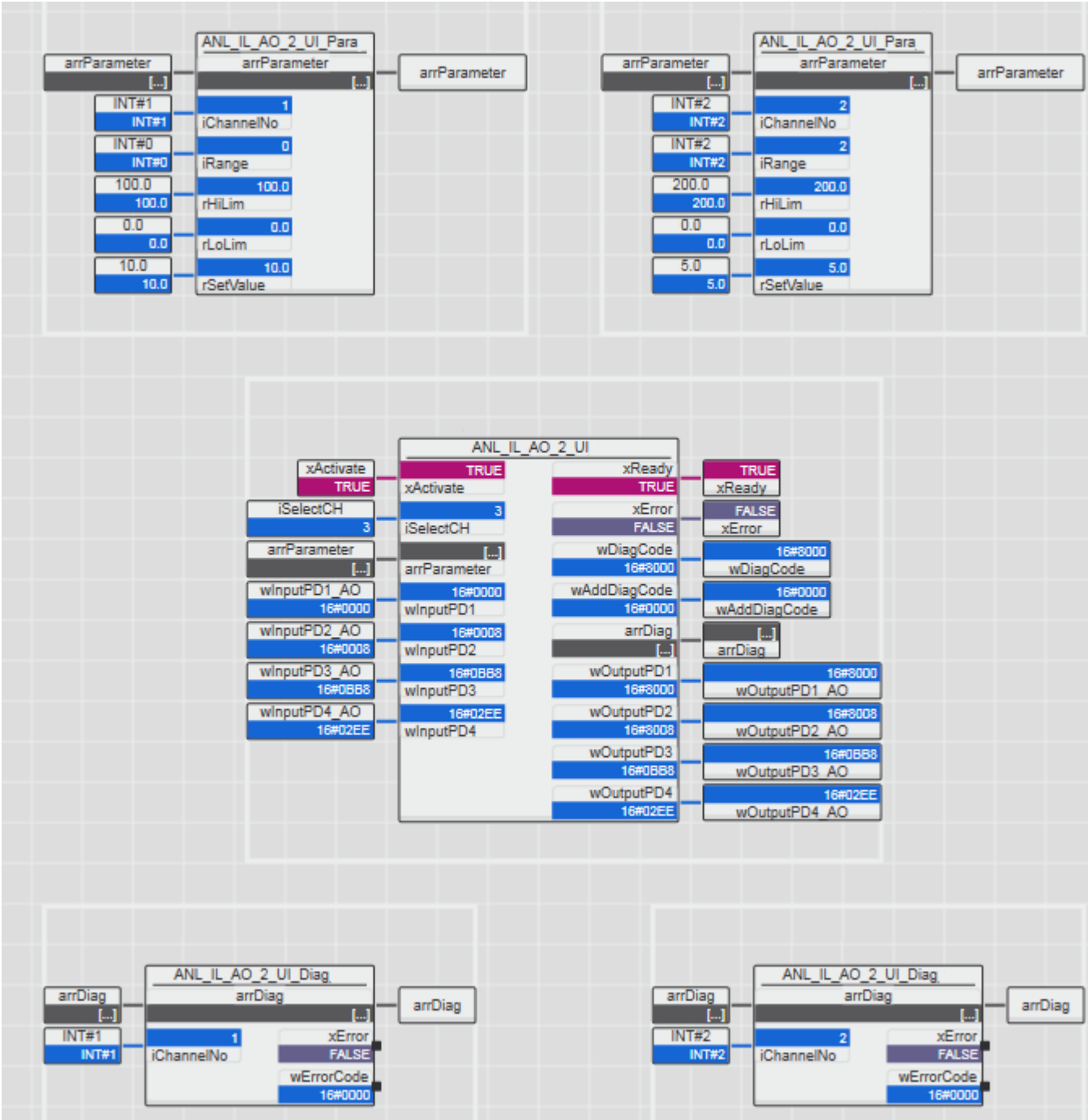
13.4.1 Bus structure

For this example, the following hardware is used:

- AXC F 2152 (2404267)
- IL PN BK DI8 DO4 2TX-PAC (2703994)
- IB IL AO 2/UI-PAC (2700775)

13.4.2 Channel parameterization with different ranges

In this example channel 1 as well as channel 2 are parameterized. For every channel the ANL_IL_AO_2_UI_Para function block is called and the inputs are provided with the required parameters. By activating the ANL_IL_AO_2_UI function block and the channel selection iSelectCh, the values of the structure arrParameter are adopted and the analog terminal is parameterized for both channels. As soon as the parameterization is finished successfully, the output parameter xReady is set to true. A changed parameterization can be set during operation at every time. In case the function blocks have faulty parameters the function block reports an error with xError and the corresponding diagnostic code. In case a hardware error occurs for channel 1 or channel 2, it is displayed differentiated in the arrDiag structure. With the ANL_IL_AO_2_UI_Diag function block the data can be read out of the structure. The output structure is assigned to the ANL_IL_AO_2_UI_Diag function block and dissolved in single output variables. For every channel the ANL_IL_AO_2_UI_Diag function block has to be called (as is the case with parameterizing) and the required analog channel has to be selected at the input iChannelNo. The block remains ready for operation despite the diagnostic message for channel 1 and channel 2.

ANL_IL_AO_2_UI_Para

| arrParameter [...] | arrParameter | arrParameter [...] | arrParameter |

INT#1 / INT#1	iChannelNo	1
INT#0 / INT#0	iRange	0
100.0 / 100.0	rHiLim	100.0
0.0 / 0.0	rLoLim	0.0
10.0 / 10.0	rSetValue	10.0

ANL_IL_AO_2_UI_Para

| arrParameter [...] | arrParameter | arrParameter [...] | arrParameter |

INT#2 / INT#2	iChannelNo	2
INT#2 / INT#2	iRange	2
200.0 / 200.0	rHiLim	200.0
0.0 / 0.0	rLoLim	0.0
5.0 / 5.0	rSetValue	5.0

ANL_IL_AO_2_UI

Input			Output	
xActivate / TRUE	xActivate	TRUE	xReady	TRUE / xReady TRUE
iSelectCH / 3	iSelectCH	3	xError	FALSE / xError FALSE
arrParameter / [...]	arrParameter	[...]	wDiagCode	16#8000 / wDiagCode 16#8000
wInputPD1_AO / 16#0000	wInputPD1	16#0000	wAddDiagCode	16#0000 / wAddDiagCode 16#0000
wInputPD2_AO / 16#0008	wInputPD2	16#0008	arrDiag	[...] / arrDiag [...]
wInputPD3_AO / 16#0BB8	wInputPD3	16#0BB8	wOutputPD1	16#8000 / wOutputPD1_AO 16#8000
wInputPD4_AO / 16#02EE	wInputPD4	16#02EE	wOutputPD2	16#8008 / wOutputPD2_AO 16#8008
			wOutputPD3	16#0BB8 / wOutputPD3_AO 16#0BB8
			wOutputPD4	16#02EE / wOutputPD4_AO 16#02EE

ANL_IL_AO_2_UI_Diag

| arrDiag [...] | arrDiag | arrDiag [...] | arrDiag |

INT#1 / INT#1	iChannelNo	1
	xError	FALSE
	wErrorCode	16#0000

ANL_IL_AO_2_UI_Diag

| arrDiag [...] | arrDiag | arrDiag [...] | arrDiag |

INT#2 / INT#2	iChannelNo	2
	xError	FALSE
	wErrorCode	16#0000

14 ANL_IL_AO4_8

The IL_AO4_8 function block is used to parameterize and standardize measured values for the IB IL AO 2 /U/BP-PAC (2861467) and IB IL AO 4/8 /U/BP 2MBD-PAC (2878052) terminal types. A plausibility check is performed on the parameter data entered with this block. Errors detected by the function block are displayed in the diagnostic register of the function block. These include exceeding parameter data limit values. A function block call is required for each Inline terminal.

14.1 Function block call
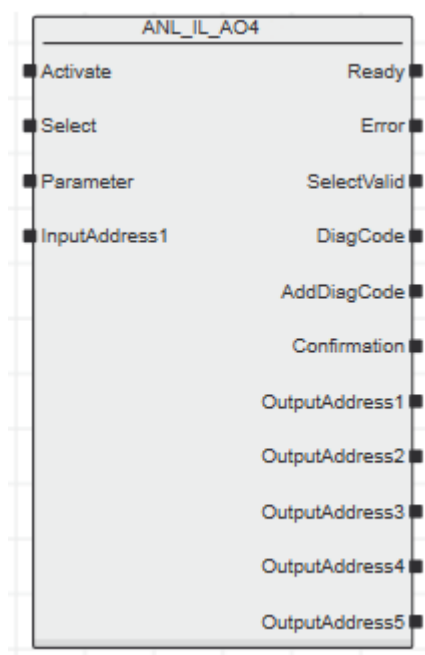
14.2 Input parameters

Name	Type	Description
Activate	BOOL	Rising edge: Activates the function block. FALSE: Deactivates the function block.
Select	BYTE	By activating the individual bits from 0 - 7 (correspond to channels 1 to 8), automatic reading in of the individual channels is activated (Multiplex mode).
Parameter	UDT_AO4_8_Para	Structure with parameter data for the terminals.
InputAddress1	WORD	Process data input address for the terminal (status message).

14.3 Output parameters

Name	Type	Description
Ready	BOOL	Function block is ready and parameterization is completed.
xError	BOOL	TRUE: An error has occurred. For details refer to wDiagCode and wAddDiagCode.
SelectValid	BOOL	Selected channels are ready.
wDiagCode	WORD	Diagnosis code. Refer to diagnostic table.
wAddDiagCode	WORD	Additional diagnosis code. Refer to diagnostic table.
wOutputAddress1	WORD	Process data output address for the terminal (commands).
wOutputAddress2	WORD	Process data output address for the terminal (parameters/output values).
wOutputAddress3	WORD	Process data output address for the terminal (parameters/output values).
wOutputAddress4	WORD	Process data output address for the terminal (parameters/output values).
wOutputAddress5	WORD	Process data output address for the terminal (parameters/output values).

14.4 Function block parameters

14.4.1 Channel parameters

The channels can be configured with the Hold and Range parameters. For the meaning of the configuration values, please refer to the data sheet. The value to be written is written to the Value parameter in real format. The maxValue and minValue limit values are used for standardization. If the value has been successfully transmitted to the module, the Done parameter = TRUE for one cycle. In the event of an error, the Error bit is assigned. Errors occur if the value to be written is greater or smaller than the limit values, or if the limit values are faulty (minValue > maxValue).

14.4.2 UDT_AO4_8_Para structure parameters

Name	Type	Description
HOLD	BOOL	Output behavior on bus reset.
Range	INT	Channel output range.
Value	REAL	Set point.
MinValue	REAL	Minimum set point.
MaxValue	REAL	Maximum set point.

14.4.3 UDT_AO4_8_Data structure confirmation

Name	Type	Description
DONE	BOOL	Set point transmitted (for one cycle set to TRUE).

14.5 Diagnosis

wDiagCode	wAddDiagCode	Description
16#0000	16#0000	Function block is deactivated.
16#8300	16#0000	Function block is in regular operation
16#C001		Timeout
	16#0000	Timeout at request to module.
16#C011		Range limit exceeded.
	16#000X	X = Channel
16#C012		MinValue > MaxValue
	16#000X	X = Channel
16#C013		Value < MinValue
	16#000X	X = Channel
16#C014		Value > MaxValue
	16#000X	X = Channel
16#C015		MinValue = MaxValue
	16#000X	X = Channel

If several errors occur simultaneously, only the first error is indicated initially. The other error messages are displayed once the first error has been removed and acknowledged. The error is reset by generating a positive edge at Activate. This process is repeated until all the errors have been removed.

14.6 Startup instructions

14.6.1 Enabling

In order to use the function block, integrate it into your application. A function block call is required for each analog terminal. Activate it after the first PLC cycle. This is done by controlling the Activate input variable from FALSE to TRUE. The channels are parameterized with a positive signal edge at Activate. As soon as this has been completed successfully, this is indicated by Ready FALSE => TRUE.

Please note that at least one cycle is required with Activate = FALSE for the internal variables to be initialized.

If the module is stopped (voltage failure, bus system stopped, etc.) it may be necessary to repeat parameterization. Activate should be controlled as described above. As soon as all the selected channels have been read in without any errors, this is indicated by the SelectValid output parameter = TRUE. If a channel is switched with an error (e.g., Value exceeds MaxValue), the status of the SelectValid output parameter = FALSE. The error cause can be determined using the DiagCode block parameter.

14.6.2 Parameterizing analog channels

In order to parameterize the channels the UDT_AO4_8 _Para parameter structure must be provided at the Parameter block parameter. The Activate input parameter must have a positive edge. It can only be reparameterized with a positive edge. The end of parameterization is indicated by Ready (FALSE => TRUE). This procedure is also required to reset an error that has been removed.

The function block performs a plausibility check on the specified parameter values. If invalid values have been specified for a parameter, the block generates an error message (DiagCode and AddDiagCode), which precisely identifies the faulty channel and parameter. The parameters are provided with 0 on initial instantiation of the block.

14.6.3 Input of standardized measured values

In order to write output values for channels 1 - 8 of an analog terminal, the Select input variable must be provided with a corresponding value.
The channels are written in order from channel 1 to 8. Therefore, the Select input parameter must be provided with a corresponding value. The activated bits in the Select parameter permit the channels to be controlled. Therefore, the least significant bit (bit 0) corresponds to channel 1, the most significant bit (bit 7) corresponds to channel 8.

Channels that are not selected are skipped. This means that it is quicker to read in the used channels.
Multiplex mode is used when reading channel 5 and onwards. It is to be expected that this will more or less double update times compared to reading the first four channels.

The validity indication for channels is output following activation of the block in the UTD_AO4_8_Conf structure of the Confirmation output parameter.
The Done variable (FALSE => TRUE) indicates that a new output value has been set for a cycle. Output values are entered in the Value variables as a real value together with the MaxValue and MinValue limit values in the Parameter input parameter. If an error is detected (e.g., limit value exceeded), the Error bit is controlled from FALSE to TRUE. DiagCode contains the error code. The breakdown of errors is described under Diagnosis.

14.7 Appendix

14.7.1 Data types

```
TYPE
    (* UDT_AO4_8_Para *)
    Analog_UDT_AO4_8_IN : STRUCT
        Hold              : BOOL; (* Output behavior on bus reset *)
        Range             : INT; (* Channel output range *)
        Value             : REAL; (* Set point *)
        MinValue          : REAL; (* Minimum se tpoint *)
        MaxValue          : REAL; (* Maximum set point *)
    END_STRUCT;
    (* UDT_AO4_8_Data *)
    Analog_UDT_AO4_8_OUT : STRUCT
        (* Set point transmitted (for one cycle set to TRUE) *)
        Done              : BOOL;

    END_STRUCT;

    UDT_IL_AO4_8_Para : ARRAY[1..8] OF Analog_UDT_AO4_8_IN;
    UDT_IL_AO4_8_Conf : ARRAY[1..8] OF Analog_UDT_AO4_8_OUT;
    AR_W_1_8 : ARRAY[1..8] OF WORD;
END_TYPE
```

15 Support

For technical support please contact your local PHOENIX CONTACT agency

at https://www.phoenixcontact.com

Owner:

PHOENIX CONTACT Electronics GmbH
Business Unit Automation Systems
System Services
Library Services