Function block libraries

Overview

for PC Worx

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 General information

Control programs are created under the PC Worx or PC WORX EXPRESS environments. The source code can then be divided into programs,blocks and functions by means of logical Program Organization Units (POU). Numerous function blocks are already contained in the editor wizard of the programming environment. Each block is provided with short description and online help. Additional blocks as extension to the input/output terminals or as technology functions are compiled into libraries by Phoenix Contactand can be added to a project if required.

The source code of a library provided by Phoenix Contact is protected, which means that it cannot be viewed. Libraries from Phoenix Contact can be provided with a setup. A compressed project for PC Worx 5 is stored on the computer during execution of the setup program, which must then be opened. When requested for the target directory for storing the library, the “Library” directory from PC Worx should be selected that is located in the “Documents and Settings/All Users/Documents/PC Worx” directory. The library is automatically converted to the programming environment version used and displayed as a project when opening with PC Worx 5 or 6. The library must be compiled once by the PC Worx version that is used.

Libraries must then be integrated into the user project via the PC Worx project tree. Select the libraries item and add the user library via the context menu.

Individual function blocks from Phoenix Contact such as those from the SQL and SNMP libraries are licensed, which means that a fee is charged for each controller on which licensed blocks are to be used. The blocks check the license stored on the controller during runtime and enable themselves or run for a limited time in demo mode.

The license is stored on the plug-in CF FLASH APPLIC A (Order No. 29 88 793) or SD FLASH APPLIC A (Order No. 29 88 816) modules. In the case of the compact ILC 130 ETH, ILC 150 ETH and ILC 150 GSM/GPRS controllers, the firmware with Version 3.50 or later contains the type APPLIC A license,which means that licensed blocks can run on them without restrictions. Please observe the specified performance data of the controllers in this performance class and check that they are suitable for your application.

2 Libraries

Library	Description	Version	License	PC Worx version
AnalogTechnology	Function blocks for acquisition and evaluation of analog signals.	8	None	From 6.30.2907
AsynCom	Function blocks for asynchronous communication for Axioline, INTERBUS and PROFINET.	4	None	From 6.30.2349
AXL_Analog	Function blocks for acquisition and evaluation of analog signals for the Axioline system.	2	None	From 6.30.2907
AXL_Basic	This function block library provides functions for devices from the Axioline product range.	1.00	None	From 6.30.601
AXL_ComSerial	Function blocks for implementing communication protocols as well as activating communication modules for the Axioline system.	3	None	From 6.30.2519
AXL_PM	Function blocks for configuring the AXL F PM EF 1F (2702671) module.	1	None	From 6.30.2349
AXL_PDI	Function blocks for the Axioline system.	2	None	From 6.30.767
AXL_SGI	Function blocks for configuring the !AXL_F_SGI2_1H module.	4	None	From 6.30.2907
Building Automation	This library is no longer supported.			
CANbus	Function blocks for support of and communication with the CANbus as well as for CAN-based protocols (CANopen, J1939, etc.).	4	None	From 6.30.1202
ComSerial	Function blocks for implementing communication protocols as well as activating communication modules.	8	None	From 6.30.1668
ComSiemens	Function blocks for communicating between controllers from Phoenix Contact and Siemens via Ethernet.	3	None	From 6.10.200
Control Technology	Function blocks for control applications.	2	Some function blocks of this library require an APPLIC-A license	From 6.30.2349
DALI_Basic	Function blocks for communication with the IB IL DALI/PWR-PAC, IB IL DALI-PAC, IB IL DALI/MM-PAC Inline modules.	4	None	From 6.30.2907
Datalogger	Function blocks for logging variables to a CSVfile. The library contains several function blocks for each data type that is supported by the data logger.	1.12	None	From 6.10.200
DBFL_SQL	Function blocks as database drivers for MS SQL and MySQL applications.	2	Some function blocks of this library require an APPLIC-A license	From 6.30.1202

Drives	Function blocks for power-level terminals from Phoenix Contact.	1.05	None	From 5
EDCL	This library is no longer supported.			
EEM	Function blocks for communication with EEM energy meters via Modbus RTU, Modbus TCP or RS232.	2	None	From 6.30.2907
FileHandling	Function blocks for handling file access.	1.00	None	From 6.10.169
FunctionModules	Function blocks for acquisition, open and closed-loop control (drivers for position detection terminals for incremental encoders, terminals with counting function or communication with IO-Link devices).	2	None	6.30.2349
HART_Basic	Function blocks for using the HART modules from Phoenix Contact.	2	None	From 6.10.200
ILCME_MCE	Motion control function blocks for ILC 191 ME.	1.02	None	From 6.30.601
ILCME_ModBus	Modbus function blocks for ILC 191 ME.	1.01	None	From 6.30.601
InterBus	INTERBUS handling.	1.11	None	From 6.10.200
InterBus_PN	INTERBUS handling on a PROFINET proxy.	2.02	None	From 6.20
IntrinsicallySafe	Function block for parameterizing and communication of modules which are used in Zone 2 according to EN 60079.	2	None	From 6.30.1914
IOL_Basic	The function block enables the asynchronous communication with the Phoenix Contact IO-Link modules.	3	None	From 6.30.2349
IT_Library	Function blocks for using IT communication protocols.	8	Some function blocks of this library require an APPLIC-A license	From 6.30.2907
IT_Security	Function blocks for safe communication.	1.00	None	From 5
IP_Com	Function blocks for establishing IP connections via a controller.	3	None	From 6.30.2907
Lighting	Function blocks for controlling PLD machine lights.	1.01	None	From 6.10.200
LMS_Modbus	Function blocks for the Lightning Monitoring System LMS.	1.00	None	From 6.30.767
MBUS	This library is no longer supported.			
MCE_Library	Function blocks for motion control.	3	None	From 6.30.1202
Modbus	This library is no longer supported.			
Modbus_RTU	Function blocks for communication with the controller via Modbus protocol Modbus RTU.	5	None	From 6.30.2907
Modbus_TCP	Function blocks for communication with the controller via Modbus protocol Modbus TCP.	4	None	From 6.30.2907

OperatorPanel	Function blocks for communication with the operator panel NLC OP2 LCD 076 4X20.	1.00	None	From 6.10.200
PDPI_Basic	P, D, PD, I, PI, PID controllers Self-tuning controllers for temperature control.	2.11	Some function blocks of this library require a PDPI-Basic license	From 6.20.331
PDPI_Pro	P, D, PD, I, PI, PID controllers Self-tuning controllers for special process automation functions.	3	Some function blocks of this library require a PDPI-Basic license	From 6.20.331
PN_Dev_Diag	Function blocks for handling PROFINET.	4	None	From 6.30.2907
Positioning	Function blocks for positioning terminals from Phoenix Contact.	1.11	None	From 6.10.200
PowerMeasurement	Function blocks for power measurement terminals from Phoenix Contact.	3	None	From 6.30.1914
PowerSupplies	Function blocks of the PowerSupplies library process data and states provided by the UPS via PROFINET.	1	None	From 6.30.767
PROFIdrive	Driver blocks for Profile Drive Technology.	2	None	From 6.30.767
RadiolineBasic	The RadiolineBasic library contains driver blocks for the currently available Radioline devices.	2	None	From 6.30.1202
RDNCY	Function blocks for handling redundancy systems.	3	None	From 6.10.200
Redundancy_ASR	Function blocks for handling redundancy systems.	1.10	None	From 6.00.25
SBT_Technology	Function blocks for a standard controller for startup of the logic modules and for exchanging data between the safe I/O modules and the logic module.	1.08	None	From 5
SBT_V3	Function blocks for a standard controller for startup of the logic modules and for exchanging data between the safe I/O modules and the logic module.	4	None	From 6.30.767
SimpleLogger	Function block for logging a maximum of nine strings each cycle.	2	None	From 6.30.2907
SNMP	Function blocks for using the Simple Network Management Protocol (SNMP v2c).	3	Some function blocks of this library require an APPLIC-A license	From 6.30.2907

SNMP3	Function blocks for using the Simple Network Management Protocol (SNMP v3).	2.00	Some function blocks of this library require an APPLIC-A license	From 5
SYS_PLC	Function block library for parameterization of Phoenix Contact PLCs.	1.21	None	From 6.30.1202
TempConversion	Function blocks for data exchange between controllers and temperature modules from Phoenix Contact.	1.10	None	From 6.30.601
TouchDisplay	Function block for parameterizing a HMI.	1.00	None	From 6.10.200
UpsIqBasic	Function blocks for reading basic parameters from uninterruptible power supplies (UPS).	1.05	None	From 6.10.200
WirelessTechnology	Function blocks for controlling Bluetooth and WLAN Ethernet adapters.	1.06	None	From 5

3 Function blocks

3.1 AnalogTechnology

Function block	Description	Version	Supported articles	License
ANL_IL_SGI_2P_EF	Function block for evaluating and parameterizing the IB IL SGI 2/P/EF-PAC (2702373) module for load cells and force tranducers.	5	IB IL SGI 2/P/EF-PAC (2702373)	none
AI_NORM	Function block for standardization of analog input values for analog modules.	1.03	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)	none

			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)	
AO_NORM	Function block for standardization of analog output values for analog modules.	2	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	none

			(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)	
IL_AI_2_SF	Function block for parameterization and control of the IB IL AI 2/SF-PAC (2861302) module.	1.02	IB IL AI 2/SF (2726285) IB IL AI 2/SF-PAC (2861302)	none
IL_AI_4_EF	Function block for parameterization and control of the IB IL AI 4/EF-PAC (2878447) module.	1.11	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

IL_AI_8	Function block for parameterization and standardization of the IB IL AI 8/SF (2727831) or IB IL AI 8/IS-PAC (2861661) module.	3	IB IL AI 8/SF (2727831) IB IL AI 8/SF-PAC (2861412) IB IL AI 8/SF (2727831) IB IL AI 8/SF-PAC (2861412) IB IL AI 8/SF-2MBD (2855648) IB IL AI 8/SF-2MBD-PAC (2862042) Only for current range: IB IL AI 8/IS (2742748) IB IL AI 8/IS-PAC (2861661)	none
IL_AO_1_SF	Function block for parameterization and control of the IB IL AO 1/SF-PAC (2861315) module.	1.00	IB IL AO 1/SF-PAC (2861315)	none
IL_AO_2	Function block for parameterization and operation of the IB IL AO 2 /SF-PAC (2863083) module.	2	IB IL AO 2 /U/BP-PAC (2861467) IB IL AO 2 /SF-PAC (2863083)	none
IL_AO_2_UI	Function block for parameterization and control of the IB IL AO 2/UI-PAC (2700775) module.	1.02	IB IL AO 2/UI-PAC (2700775)	none
IL_AI_4_UI	Function block for parameterization and standardization IB IL AI 4 /I-PAC (2700458) or IB IL AI 4 /U-PAC (2700459) module.	1.04	IB IL AI 4 /I-PAC (2700458) IB IL AI 4 /U-PAC (2700459)	none
IL_AO4_8	Function block for parameterization and standardization of the IB IL AO 4/8/U/BP-PAC (2878036) module.	1.12	IB IL AO 4/8/U/BP-PAC (2878036) IB IL AO 4/8 /U/BP 2MBD-PAC (2878052)	none
IL_SGI_1_CAL	Function block for control and parameterization of the IB IL SGI 1 /CAL (2700064) module.	1.01	IB IL SGI 1 /CAL (2700064)	none

IL_SGI_2F	Function block for control and parameterization of the IB IL SGI 2/F-PAC (2878638) module.	1.00	IB IL SGI 2/F-PAC (2878638) IB IL SGI 2/F-2MBD-PAC (2878735)	none
IL_SGI_2P_A	Function block for evaluating and parameterization of the IB IL SGI 2/P-PAC (2884907) module for load cells and transducers.	1.01	IB IL SGI 2/P-PAC (2884907)	none

3.2 AsynCom

Function block	Description	Version	Supported articles	License
AsynCom_AXL	Function block for asynchronous communication with Axioline devices.	3	Refer to “Supported PLCs”	none
AsynCom_IBS	Function block for asynchronous communication with INTERBUS devices.	2	"	none
AsynCom_PN	Function block for asynchronous communication with PROFINET-devices.	2	"	none
AsynCom_PN_Info	The function block reads out the list of Node IDs and F destination addresses of all PROFINET IO Devices including the IDs of the slots.	2	"	none
AsynCom_PN_Get_Idx	The function block searches for the I/O Device (Bus coupler) index of the given Node ID in the in/output arrDeviceAsynCom.	1.00	"	none
AsynCom_F_2_NodeID	The function block searches for the Node ID of the given F-Destination-Address in the in/output arrDeviceAsynCom.	1.01	"	none

3.3 AXL_Analog

Function block	Description	Version	Supported articles	License
AXL_Analog_IN	Function block for calculating standardized data values from the process data of the analog input terminals in the Axioline product range.	2	AXL AI 8 (2688064) AXL F AI2 AO2 1H (2702072) AXL F AI4 I 1H (2688491) AXL F AI4 U 1H (2688501) AXL SE AI4 I 4-20 (1088062) AXL SE AI4 U 0-10 (1088104)	none
AXL_Analog_OUT	Function block for calculating standardized data values from the process data of the analog output terminals in the Axioline product range.	2	AXL AO8 (2688080) AXL F AI2 AO2 1H (2702072) AXL F AO4 1H (2688527) AXL SE AO4 I 4-20 (1088123) AXL SE AO4 U 0-10 (1088126)	none
AXL_RTD	Function block for calculating standardized data values for temperature recording from the process data of the input terminals in the Axioline product range.	2	AXL F RTD8 1F (2688077) AXL RTD 8-ME (2688190) AXL SE RTD4 PT100 (1088106)	none

3.4 AXL_Basic

Function block	Description	Version	Supported articles	License
AXB_HotSwap	Parameterization and restart of the local bus after module replacement in the Axioline local bus	1.00	AXC 1050 (2700988)	none

3.5 AXL_ComSerial

Function block	Description	Version	Supported articles	License
AXL_RSUNI_PD	Function block for performing the send and receive operations via the AXL F RS UNI 1H (2688666) module.	2	AXL F RS UNI 1H (2688666) AXL SE RS485 (1088128)	none

3.6 AXL_PM

Function block	Description	Version	Supported articles	License
AXL_PM_AddInfo	This function block is used to read out additional information. It uses the AsynCom library to get the objects and reads them on request.	1	AXL F PM EF 1F (2702671)	none
AXL_PM_Config	This function block is used to configurate the module.	1	AXL F PM EF 1F (2702671)	none
AXL_PM_Main	This function block processes the input process data and submits them to the AXL_PM_Values function block. It also controls the EnergyCountControl object of the module.	1	AXL F PM EF 1F (2702671)	none
AXL_PM_Values	This function block shows the measured values for the selected phase.	1	AXL F PM EF 1F (2702671)	none

3.7 AXL_PDI

Function block	Description	Version	Supported articles	License
AXL_PDI_R_W	The function block writes the required data to the udtAsynCom structure or reads the required data out of the udtAsynCom structure, so that the respective AsynCom block can communicate.	2	–	none
AXL_PDI_AddLabel	Function block for electronic rating plates.	1.01	–	none
AXL_PDI_Diag	Function block for diagnostics.	1.01	–	none

3.8 AXL_SGI

Function block	Description	Version	Supported articles	License
AXL_SGI_Main	This function block initializes the module, processes the input process data and submits them to the AXL_SGI_Values function block. The function block also writes data in the output process data.	1	AXL F SGI2 1H (2702911)	none
AXL_SGI_Control	This function block is used to adjust the module and to read additional vlaues with the AsynCom.	1	AXL F SGI2 1H (2702911)	none
AXL_SGI_Config	This function block is used to configure the module.	3	AXL F SGI2 1H (2702911)	none
AXL_SGI_Values	This function block shows the measured values for the selected channel.	2	AXL F SGI2 1H (2702911)	none

3.9 CANbus

Block	Description	Version	Article
AXL_CAN_COMM	Driver for AXL F IF CAN 1H (2702668) module	3	AXL F IF CAN 1H (2702668)
AXL_CAN_Para	Function block for parameterization of the AXL F IF CAN 1H (2702668) module	2	AXL F IF CAN 1H (2702668)
AXL_CAN_Para11	Function block for parameterization of the AXL F IF CAN 1H (2702668) module. For filter values in case of usage of 11 bit CAN identifier	2	AXL F IF CAN 1H (2702668)
AXL_CAN_Para29	Function block for parameterization of the AXL F IF CAN 1H (2702668) module. For filter values in case of usage of 29 bit CAN identifier	2	AXL F IF CAN 1H (2702668)
IL_DN_RW	Function block for communication between a DeviceNet network with available DeviceNet nodes.	1.00	IB IL CAN-MA-PAC (2700196)
IL_NMEA_RD_Multi	Function block for targeted reading of the data of a packet from a multi-packet message.	1.00	IB IL CAN-MA-PAC (2700196)
IL_NMEA_RD	Function block for reading the current values from the array of a parameter group.	1.00	IB IL CAN-MA-PAC (2700196)
IL_NMEA_WR	Function block that can make up to 8 bytes of data in one node in an NMEA network available to a node in another NMEA network by entering a CAN ID.	1.00	IB IL CAN-MA-PAC (2700196)
IL_CAN_COMM	Function block for establishing the connection to the IB IL CAN-MA-PAC (2700196)-PAC module.	2	IB IL CAN-MA-PAC (2700196)
IL_CO_DEV_Index	Function block for defining an index (0000-FFFF). So data can be changed between Master and Device (ILC).	1.00	IB IL CAN-MA-PAC (2700196)
IL_CO_DEV_Main	This function block is used as CANopen device. Supported are 10 RPDOs and 10 TPDOs. Heartbeat and Node guarding are used for monitoring.	1.10	IB IL CAN-MA-PAC (2700196)
IL_CO_DEV_SubIndex	Function block for defining a subindex (1-5). So data can be changed between Master and Device (ILC).	1.00	IB IL CAN-MA-PAC (2700196)
IL_CO_EMCY	This function block is waiting for an emergency message. Additional information regarding the emergency message can be obtained from the outputs.	1.00	IB IL CAN-MA-PAC (2700196)

IL_CO_NMT_Guard	Function block for changing operating mode of a node.	1.00	IB IL CAN-MA-PAC (2700196)
IL_CO_NMT	Function block for determination and configuration operating mode of the CANopen node.	1.00	IB IL CAN-MA-PAC (2700196)
IL_CO_NodeGuard	Function block for displaying the current operating mode of a node.	1.00	IB IL CAN-MA-PAC (2700196)
IL_CO_NodeInfo	Function block for reading information from a node about Hardware version and Software version of the module, name of the module or serial number of the module.	1.00	IB IL CAN-MA-PAC (2700196)
IL_CO_PDO_RD	Function block for receiving PDO messages (e.g., 180 or 700).	1.00	IB IL CAN-MA-PAC (2700196)
IL_CO_PDO_WR	Function block for sending PDO messages.	1	IB IL CAN-MA-PAC (2700196)
IL_CO_RD_WR	Function block for setting the objects (indexes, subindexes) of a CANopen node.	1.00	IB IL CAN-MA-PAC (2700196)
IL_CO_SDO_RD	Function block for reading contents of an index.	1.00	IB IL CAN-MA-PAC (2700196)
IL_CO_SDO_WR	Function block for assigning a new value to an index.	1.00	IB IL CAN-MA-PAC (2700196)
IL_CO_Search	Searches for available nodes in a CANopen network and displays their device names and node IDs.	1.00	IB IL CAN-MA-PAC (2700196)
IL_CO_SYNC	Function block for sending a COB-ID 80 synchronization message.	1.00	IB IL CAN-MA-PAC (2700196)
CAN_TO_AXL_STRUCT	Function block for mapping data from the CN_udt_RxTx structure to the CAN_UDT_DATA structure.	1	IB IL CAN-MA-PAC (2700196) AXL F IF CAN 1H (2702668)

CAN_TO_IL_STRUCT	Function block for mapping data from the CAN_UDT_DATA structure to the CN_udt_RxTx structure.	1	IB IL CAN-MA-PAC (2700196) AXL F IF CAN 1H (2702668)
IL_J1939_RD_Multi	Function block for reading the current data of a packet (parameter group) from a multi-packet message.	1.03	IB IL CAN-MA-PAC (2700196)
IL_J1939_READ	Function block for reading the current data of a packet (parameter group) from a standard message.	1.03	IB IL CAN-MA-PAC (2700196)
IL_J1939_WRITE	Function block for transmitting data to a node in a J1939 network.	1.03	IB IL CAN-MA-PAC (2700196)

3.10 ComSerial

Function block	Description	Version	Supported articles	License
IL_RS232	Function blocks for controlling and configuring the IB IL RS232 PCP terminal.	1.10	IB IL RS 232 (2727349) IB IL RS 232-PAC (2861357)	none
IL_RS232P	Function blocks for controlling and configuring the IB IL RS232 PRO terminal.	4	IB IL RS 232-PRO (2878515) IB IL RS 232-PRO-PAC (2878722)	none
IL_RS485	Function blocks for controlling and configuring the IB IL RS485 PCP terminal.	1.10	IB IL RS 485/422 (2836793) IB IL RS 485/422-PAC (2861933)	none
IL_RS485P	Function blocks for controlling and configuring the IB IL RS485 PRO terminal.	3	IB IL RS 485/422-PRO (2863707) IB IL RS 485/422-PRO-PAC (2863627)	none
IL_RSUNI	Function blocks for controlling and configuring the IB IL RS UNI terminal.	6	IB IL RS UNI-PAC (2700893)	none
IL_RS485_ECO	Function blocks for controlling and configuring the IB IL RS 485-ECO terminal.	7	IB IL RS 485-ECO (2702795)	none
IL_RS232_ECO	Function blocks for controlling and configuring the IB IL RS 232-ECO terminal.	5	IB IL RS 232-ECO (2702141)	none

3.11 ComSiemens

Function block	Description	Version	Supported articles	License
CoSi_ETH_S7_Link	This block manages the TCP/IP connection between the Phoenix PLC and the Siemens PLC.	1.02	–	none
CoSi_GetSetDint	Set function writes a DINT value into the data buffer. Get function reads a DINT value out of the data buffer.	1.00	–	none
CoSi_GetSetDword	Set function writes a DWORD value into the data buffer. Get function reads a DWORD value out of the data buffer.	1.00	–	none
CoSi_GetSetInt	Set function writes an INT value into the data buffer. Get function reads an INT value out of the data buffer.	1.00	–	none
CoSi_FIFO	This block implements a FIFO (First In First Out) buffer. Data can be stored in a byte array and read out again.	1.00	–	none
CoSi_GetSetReal	Set function writes a REAL value into the data buffer. Get function reads a REAL value out of the data buffer.	1.00	–	none
CoSi_GetSetString	Function block for reading and writing a string to the data buffer of the ETH_S7Link block.	1.00	–	none
CoSi_GetSetTime	Set function converts a TIME value into the S5TIME data type and writes it into the data buffer. Get function reads an S5TIME value out of the data buffer and converts it into the TIME data type.	1.00	–	none
CoSi_GetSetWord	Set function writes a WORD value into the data buffer. Get function reads a WORD value out of the data buffer.	1.00	–	none
CoSi_JobInterface	This block provides an interface for processing several requests one after the other via the CoSi_JobManager block.	3	–	none
CoSi_JobManager	This block manages the requests that are sent to the CoSi_ETH_S7_Link block and processes them one after the other.	2	–	none

3.12 ControlTechnology

Function block	Description	Version	Supported articles	License
SC_W_R	Scaling of the analog input value and type conversion from WORD to REAL.	2	-	none
SC_R_W	Scaling of the analog output value and type conversion from REAL to WORD.	2	-	none
SC_R_R	Scaling of any quanity of the type REAL.	1.02	-	none
LTR	Linear transformation.	1.01	-	none
A2_OF_A3	Analog value selection 2 out of 3.	1.01	-	none
LIMITVAL	Amplitude limiter.	1.01	-	none
LIMITROC	Rate of change limiter.	1.02	-	none
ALARM_2Q	Limit value indicator with 2 alarm limits.	1.03	-	APPLIC_A
ALARM_4Q	Limit value indicator with 2 warning and 2 alarm limits.	1.03	-	APPLIC_A
B2_OF_B3	Binary value selection 2 out of 3.	1.01	-	none
TWIN_DRIVE	Simultaneous driving of two actuators.	1.01	-	none
THREE	Three-position control element (two-position control element).	1.01	-	none
REV_LOCK	Reversing interlock.	1.01	-	none
POLG_N	Polygonal line.	1.03	-	APPLIC_A
POLN_N	Polynomial.	1.03	-	APPLIC_A
DEADBAND	Dead band (without hysteresis).	1.01	-	none
INT_C	Integrator.	1.03	-	APPLIC-A
LAG1ST	PT1 element / PT1 filter.	1.03	-	APPLIC-A
DYN	Dynamic element.	1.03	-	APPLIC-A
DELAY20	Dead time element (dead time = max. 20 x cycle time).	1.02	-	none
DELAY100	Dead time element (dead time = max. 100 x cycle time).	1.02	-	none
HOLD	Holding element.	1.01	-	none
C_N	Keying controller of the n-th order.	1.03	-	APPLIC-A
PID_C	Continuous PID-type controller.	1.23	-	APPLIC-A
PID_R	Continuous PID-type controller (with reduced performance range).	1.04	-	APPLIC-A
THREE_C	Three-position controller attachment for PID_C and PID_R.	1.12	-	none
PID_ADA	PID-type controller attachment module for controlled adaption.	1.01	-	none
PID_STR	PID-type controller attachment module for controlled adaption.	1.01	-	none
PID_MAN	PID-type controller attachment module for the input of manual manipulated values.	1.01	-	none
PID_MODE	PID-type controller attachment module for the input of the MODE control commands.	1.01	-	none
PID_PAR	PID-type controller attachment module for the input of parameterizing values.	1.01	-	none
PID_STAT	PID-type controller attachment module for the output of status information.	1.01	-	none

CYCLE	Calculation of the current cycle time.	1.01	-	none
DRIVE_SIM	Simulation of a servo motor.	1.01	-	none
SEND50	Storage of 50 successive values.	1.01	-	none
RCV50	Display of the 50 determined values in FBD.	1.01	-	none
ADA_PAR	Clear parameter transfer in FBD.	1.01	-	none
AG_PAR	Clear parameter transfer in FBD.	1.01	-	none
C_N_PAR	Simplified parameter setting for the function block C_N.	1.01	-	none
POL_PAR	Simplified parameter setting for the function block POLN_N.	1.01	-	none
MODE_PAR	Simplified parameter setting for the function block PID_MODE.	1.01	-	none
PAR_PAR	Simplified parameter setting for the function block PID_PAR.	1.01	-	none

3.13 DALI_Basic

Function block	Description	Version	Supported articles	License
AXL_DALI_InputPD	The AXL_DALI_InputPD does a mapping of the AXL F MA DALI2 1H (2702864) input process data to the DALI_Server function block input process data.	1	AXL F MA DALI2 1H (2702864)	none
AXL_DALI_OutputPD	The AXL_DALI_OutputPD does a mapping of the DALI_Server function block output process data to the AXL F MA DALI2 1H (2702864) output process data.	2	AXL F MA DALI2 1H (2702864)	none
DALI_Server	Function block for communicating with the IB IL DALI/PWR-PAC (2897813) and IB IL DALI-PAC (2897910) modules or AXL F MA DALI2 1H (2702864).	2	AXL F MA DALI2 1H (2702864) IB IL DALI/PWR-PAC (2897813) IB IL DALI-PAC (2897910)	none
DALI_MM_Server	Function block for communicating with the IB IL DALI/MM-PAC (2700605) modules.	3	IB IL DALI/MM-PAC (2700605)	none
DALI_MM_SensorM2	Function block that works with a MM server block to record the brightness measured by the MSensor02 sensor and to detect a presence.	1.01	–	none
DALI_HystControl	Function block that works with the M2 sensor and the DimGroup block to regulate the brightness of a room.	2	–	none
DALI_DimT1	Function block for intuitively switching and dimming lights.	1.00	–	none
DALI_DimGroup	Function block that works with a server block to dim a light group.	1.00	–	none
DALI_ArcPGroup	Function block for controlling the light intensity of a light group.	1.00	–	none
DALI_OnOffGroup	Function block for switching a light group on and off.	1.00	–	none
DALI_SceneGroup	Function block for setting a light scene for a light group.	1.00	–	none
DALI_Group	Function block that logically combines the DimT1, the DimGroup, the ArcPGroup, the OnOffGroup, and the SceneGroup blocks together.	1.01	–	none
DALI_Groups	Function block that contains and combines the 16 DALI_Group blocks.	1.01	–	none
DALI_BalastT1	Works with a server block to determine and output the parameters of a light. Changes to the parameters can be made directly at the block.	2	–	none
DALI_State	Function block for determining the state of all 64 possible devices.	1.01	–	none

3.14 Datalogger

Function block	Description	Version	Supported articles	License
DataLogger	This function block is used to collect data and convert into a buffer in ASCII format.	1.00	-	none
DataLogBool	This function block is used to log variables of the data type BOOL.	1.00	-	none
DataLogByte	This function block is used to log variables of the data type BYTE.	1.00	-	none
DataLogDint	This function block is used to log variables of the data type DINT.	1.00	"	none
DataLogDword	This function block is used to log variables of the data type DWORD.	1.00	"	none
DataLogInt	This function block is used to log variables of the data type INT.	1.00	"	none
DataLogLreal	This function block is used to log variables of the data type LREAL.	1.00	"	none
DataLogReal	This function block is used to log variables of the data type REAL.	1.00	"	none
DataLogSint	This function block is used to log variables of the data type SINT.	1.00	"	none
DataLogUdint	This function block is used to log variables of the data type UDINT.	1.00	"	none
DataLogUint	This function block is used to log variables of the data type UINT.	1.00	"	none
DataLogUsint	This function block is used to log variables of the data type USINT.	1.00	"	none
DataLogWord	This function block is used to log variables of the data type WORD.	1.00	"	none
DataLogCSV	This function block is used to store data to the local file system as CSV files.	1.00	"	none
DataLogFTP	This FB is used to log data to files on FTP Server.	1.00	"	none
DataLogMsSql	This FB is used to log data into SQL data base.	1.00	"	none
DI_TimeFormat	This function block is used to generate different time-formats.	1.00	"	none
DataLogOnEvent	This function block is used to record each cycle one data set in the RAM.	1.00	"	none
DataLogEvCSV	This function block is used to store the logged data from the function block “DataLogOnEvent”.	1.00	"	none

3.15 DBFL_SQL

Function block	Description	Version	Supported articles	License
DBFL_TSQL_ACCESS	The function block enables access to the MsSQL database.	1.16	-	Applic-A licence
DBFL_MySQL_ACCESS	The function block enables access to the MySQL / MariaDB database.	1.15	-	Applic-A licence
DBFL_TSQL_DECODE	The function block is used to evaluate a received table and is used as a continuation block of DBFL_TSQL_ACCESS.	1.16	-	none
DBFL_MySQL_DECODE	The function block is used to evaluate a received table and can be used as a continuation block of DBFL_MySQL_ACCESS.	2	-	none
DBFL_CommandFiFo	The function block saves up to 50 SQL commands.	1.02	-	none
DBFL_CODE	The function block inserts SQL commands or parts of them in the "SQL_OUT" array.	1.01	-	none
DBFL_StartComT1	The function block creates the start of a database command.	1.00	-	none
DBFL_StartComT2	The function block creates the start of a database command.	1.01	-	none
DBFL_BoolToComT1	The function block inserts a Boolean value in the SQL command.	1.00	-	none
DBFL_BoolToComT2	The function block inserts a Boolean value in the SQL command.	1.01	-	none
DBFL_IntToComT1	The function block inserts an integer value in the SQL command.	1.00	-	none
DBFL_IntToComT2	The function block inserts an integer value in the SQL command.	1.01	-	none
DBFL_DIntToComT1	The function block inserts a DINT value in the SQL command.	1.00	-	none
DBFL_DIntToComT2	The function block inserts a DINT value in the SQL command.	1.01	-	none
DBFL_RealToComT1	The function block inserts a REAL value in the SQL command.	1.00	-	none
DBFL_RealToComT2	The function block inserts a REAL value in the SQL command.	1.01	-	none
DBFL_DateTimeStrT1	The function block inserts a date/time value in the SQL command.	1.02	-	none
DBFL_DateTimeStrT2	The function block inserts a date/time value in the SQL command.	1.02	-	none
DBFL_ByteToComT1	The function block inserts a byte value in the SQL command.	1.00	-	none
DBFL_ByteToComT2	The function block inserts a byte value in the SQL command.	1.01	-	none
DBFL_WordToComT1	The function block inserts a data word in the SQL command.	1.00	-	none
DBFL_WordToComT2	The function block inserts a data word in the SQL command.	1.01	-	none

DBFL_StrToComT1	The function block inserts a string in the SQL command.	1.00	-	none
DBFL_StrToComT2	The function block inserts a string in the SQL command.	1.01	-	none

3.16 Drives

Function block	Description	Version	Supported articles	License
IL_MOT400	Function block for parameterizing, diagnosing, and controlling Inline 400 motor starters.	1.01	IB IL 400 MLR 1-8A (2727365) IB IL 400 MLR 1 8A 2MBD (2855428) IB IL 400 ELR 1-3A (2727352) IB IL 400 ELR 1 3A 2MBD (2855525) IB IL 400 ELR R 3A (2727378) IB IL 400 ELR R 3A 2MBD (2855130)	none
IL_MLR_R	Function block for parameterizing and controlling motor starters in reversing-load operation.	1.01	IB IL 400 MLR 1-8A (2727365) IB IL 400 MLR 1 8A 2MBD (2855428)	none

3.17 EEM

Function block	Description	Version	Supported articles	License
EEM_0000_RTU	Function block for reading and standardizing instantaneous values of register addresses 16#0000 – 16#0052.	1	EEM-EM355 (2908578) EEM-EM357 (2908588)	none
EEM_0370_RTU	Function block for reading and standardizing values of register addresses 16#0370 - 16#0374. The register addresses display current average current/voltage value based on current and voltage transducer (under consideration of transmission ratios).	1	EEM-MA600 (2901366)[*2],[*10]	none
EEM_0741_RTU	Function block for reading and standardizing values of register addresses 16#0741 – 16#0743. The register addresses display current average current/voltage value not based on current and voltage transducer (without consideration of transmission ratios).	1	"	none
EEM_1000_RTU	Function block for reading and standardizing instantaneous values of register addresses 16#1000 – 16#103A.	1	EEM-EM355 (2908578) EEM-EM357 (2908588)	none
EEM_1100_RTU	Function block for reading and standardizing meter values of register addresses 16#1100 – 16#114E.	1	"	none
EEM_8000_RTU	Function block for reading and standardizing instantaneous values of register addresses 16#8000 – 16#8057.	1	EEM-MA250 (2901363)[*10] EEM-MA370 (2907983) EEM-MA371-R (29079850) EEM-MA400 (2901364)[*1],[*10] EEM-MA600 (2901366)[*2],[*10] EEM-MA600-24DC (2902352)[*2],[*10] EEM-MA770-R (2907944) EEM-MA771-R (2908285)	none
EEM_8100_RTU	Function block for reading and standardizing energy meter values of register addresses 16#8100 – 16#8116.	1	"	none
EEM_8200_RTU	Function block for reading and standardizing tariff meter values of register addresses 16#8200 – 16#822F.	1	"	none
EEM_8300_RTU	Function block for reading and standardizing operating hours of register addresses 16#8300 – 16#8302.	1	"	none

EEM_8340_RTU	Function block for reading and standardizing the state of impulse counter of digital input of register address 16#8340.	1	"	none
EEM_8500_RTU	Function block for reading and standardizing statistics (prediction) of register addresses 16#8500 - 16#850E.	1	"	none
EEM_8540_RTU	Function block for reading and standardizing statistics (mean values) of register addresses 16#8540 - 16#855E.	1	"	none
EEM_8600_RTU	Function block for reading and standardizing min / max values for mean values of register addresses 16#8600 - 16#863E.	1	"	none
EEM_8700_RTU	Function block for reading and standardizing min / max values for instantaneous values of register addresses 16#8700 - 16#873E.	1	"	none
EEM_8800_RTU	Function block for reading and standardizing total harmonic distortion of register addresses 16#8800 - 16#8809.	1	"	none
EEM_C550_RTU	Function block for reading and standardizing values of register addresses 16#C550 – 16#C58C. The register addresses display the current main measured values based on current and voltage transducer (under consideration of transmission ratios).	1	EEM-MA250 (2901363)[*10] EEM-EM355 (2908578) EEM-EM357 (2908588) EEM-MA400 (2901364)[*1],[*10] EEM-MA600 (2901366)[*2],[*10] EEM-MA600-24DC (2902352)[*2],[*10]	none
EEM_C650_RTU	Function block for reading and standardizing values of register addresses 16#C650 - 16#C690. The register addresses display energy values.	1	"	none
EEM_C750_RTU	Function block for reading and standardizing values of register addresses 16#C750 - 16#C795. The register addresses display average values based on current and voltage transducer via the set measurement duration (under consideration of transmission ratios).	1	EEM-MA250 (2901363)[*10] EEM-MA400 (2901364)[*1],[*10] EEM-MA600 (2901366)[*2],[*10] EEM-MA600-24DC (2902352)[*2],[*10]	none
EEM_C850_RTU	Function block for reading and standardizing values of register addresses 16#C850 - 16#C872. The register addresses display current main measured values not based on current and voltage transducer (without consideration of transmission ratios).	1	"	none
EEM_C900_RTU	Function block for reading and standardizing values of register addresses 16#C900 - 16#C907. The register addresses display temperature inputs.	1	"	none

EEM_C950_RTU	Function block for reading and standardizing values of register addresses 16#C950 - 16#C959. The register addresses display current harmonic distortion and harmonics.	1	"	none
EEM_C95A_RTU	Function block for reading and standardizing values of register addresses 16#C95A - 16#CA92. The register addresses display individual harmonic content for currents.	1	EEM-MA600 (2901366) [*2],[*10] EEM-MA600-24DC (2902352) [*2],[*10]	none
EEM_CB00_RTU	Function block for reading and standardizing values of register addresses 16#CB00 - CB03. The register addresses display the status of inputs/outputs.	1	EEM-MA250 (2901363)[*10] EEM-MA400 (2901364)[*1],[*10] EEM-MA600 (2901366)[*2],[*10] EEM-MA600-24DC (2902352)[*2],[*10]	none
EEM_0000_TCP	Function for reading and standardizing instantaneous values of register addresses 16#0000 – 16#0052.	1	EEM-MA375 (2908581) EEM-MA377 (2908590)	none
EEM_0370_TCP	Function block for reading and standardizing current average current/voltage value based on current and voltage transducer (under consideration of transmission ratios) of register addresses 16#0370 – 16#0374.	1	EEM-MA600 (2901366)[*2],[*10]	none
EEM_0741_TCP	Function block for reading and standardizing current average current/voltage value not based on current and voltage transducer (without consideration of transmission ratios) of register addresses 16#0741 – 16#0743.	1	"	none
EEM_1000_TCP	Function block for reading and standardizing instantaneous values of register addresses 16#1000 – 16#103A.	1	EEM-MA375 (2908581) EEM-MA377 (2908590)	none
EEM_1100_TCP	Function block for reading and standardizing meter values of register addresses 16#1100 – 16#114E.	1	"	none

EEM_8000_TCP	Function block for reading and standardizing instantaneous values of register addresses 16#8000 – 16#8057.	1	EEM-MA370 (2907983) EEM-MA370-R (2907980) EEM-MA371 (2908307) EEM-MA371-R (29079850) EEM-MA600 (2901366)[*3],[*10] EEM-MA600-24DC (2902352)[*3],[*10] EEM-MA770 (2907945) EEM-MA770-R (2907944) EEM-MA770-EIP (2907953) EEM-MA770-PN (2907946) EEM-MA771 (2908286) EEM-MA771-R (2908285) EEM-MA771-EIP (2908302) EEM-MA771-PN (2908301) EEM-MB370 (2907954) EEM-MB370-EIP (2907971) !EEM_MB370_PN EEM-MB371 (2907955) EEM-MB371-EIP (2907976) EEM-MB371-PN (2908308)	none
EEM_8100_TCP	Function block for reading and standardizing energy meter values of register addresses 16#8100 – 16#8116.	1	"	none
EEM_8200_TCP	Function block for reading and standardizing tariff meter values of register addresses 16#8200 – 16#822F.	1	"	none
EEM_8300_TCP	Function block for reading and standardizing operating hours of register addresses 16#8300 – 16#8302.	1	"	none
EEM_8340_TCP	Function block for reading and standardizing the state of impulse counter of digital input of register address 16#8340.	1	"	none
EEM_8500_TCP	Function block for reading and standardizing statistics (prediction) of register addresses 16#8500 - 16#850E.	1	"	none

EEM_8540_TCP	Function block for reading and standardizing statistics (mean values) of register addresses 16#8540 - 16#855E.	1	"	none
EEM_8600_TCP	Function block for reading and standardizing min / max values for mean values of register addresses 16#8600 - 16#863E.	1	"	none
EEM_8700_TCP	Function block for reading and standardizing min / max values for instantaneous values of register addresses 16#8700 - 16#873E.	1	"	none
EEM_8800_TCP	Function block for reading and standardizing total harmonic distortion of register addresses 16#8800 - 16#8809.	1	"	none
EEM_C550_TCP	Function block for reading and standardizing the data in Table 16#C550 (register addresses C550 – C58C) from EMPros 250, 400 and 600. The register addresses display the current main measured values based on current and voltage transducer (under consideration of transmission ratios).	1	EEM-MA600 (2901366)[*3],[*10] EEM-MA600-24DC (2902352)[*3],[*10]	none
EEM_C650_TCP	Function block for reading and standardizing the data in Table 16#C650 (register addresses C650 – C690) from EMPros 250, 400 and 600. The register addresses display energy values.	1	"	none
EEM_C750_TCP	Function block for reading and standardizing the data in Table 16#C750 (register addresses C750 – C795) from EMPros 250, 400 and 600. The register addresses display average values based on current and voltage transducer via the set measurement duration (under consideration of transmission ratios).	1	"	none
EEM_C850_TCP	Function block for reading and standardizing the data in Table 16#C850 (register addresses C850 – C872) from EMPros 250, 400 and 600. The register addresses display current main measured values not based on current and voltage transducer (without consideration of transmission ratios).	1	"	none
EEM_C900_TCP	Function block for reading and standardizing the data in Table 16#C900 (register addresses C900 – C907) from EMPros 250 and 600. The register addresses display temperature inputs.	1	"	none
EEM_C950_TCP	Function block for reading and standardizing the data in Table 16#C950 (register addresses C950 – C959) from EMPros 250, 400 and 600. The register addresses display current harmonic distortion and harmonics.	1	"	none
EEM_C95A_TCP	Function block for reading and standardizing the data in Table 16#C95A (register addresses C95A – CA92) from EMPros 600. The register addresses display individual harmonic content for currents.	1	"	none

EEM_CB00_TCP	Function block for reading and standardizing the data in Table 16#CB00 (register addresses CB00 – CB03) from EMPros 250, 400 and 600. The register addresses display the status of inputs/outputs.	1	"	none
EEM_0000_Values	Function block that converts the EEM_UDT_0000 input structure into individual output parameters.	1	-	none
EEM_0370_Values	Function block that converts the EEM_UDT_0370 input structure into individual output parameters.	1	-	none
EEM_0741_Values	Function block that converts the EEM_UDT_0741 input structure into individual output parameters.	1	-	none
EEM_1000_Values	Function block that converts the EEM_UDT_1000 input structure into individual output parameters.	1	-	none
EEM_1100_Values	Function block that converts the EEM_UDT_Tab1100 input structure into individual output parameters.	1	-	none
EEM_8000_Values	Function block that converts the EEM_UDT_8000 input structure into individual output parameters.	1	-	none
EEM_8100_Values	Function block that converts the EEM_UDT_8100 input structure into individual output parameters.	1	-	none
EEM_8200_Values	Function block that converts the EEM_UDT_8200 input structure into individual output parameters.	1	-	none
EEM_8300_Values	Function block that converts the EEM_UDT_8300 input structure into individual output parameters.	1	-	none
EEM_8340_Values	Function block that converts the EEM_UDT_8340 input structure into individual output parameters.	1	-	none
EEM_8500_Values	Function block that converts the EEM_UDT_8500 input structure into individual output parameters.	1	-	none
EEM_8540_Values	Function block that converts the EEM_UDT_8540 input structure into individual output parameters.	1	-	none
EEM_8600_Values	Function block that converts the EEM_UDT_8600 input structure into individual output parameters.	1	-	none
EEM_8700_Values	Function block that converts the EEM_UDT_8700 input structure into individual output parameters.	1	-	none
EEM_8800_Values	Function block that converts the EEM_UDT_8800 input structure into individual output parameters.	1	-	none
EEM_C550_Values	Function block that converts the EEM_UDT_C550 input structure into individual output parameters.	1	-	none

EEM_C650_Values	Function block that converts the EEM_UDT_C650 input structure into individual output parameters.	1	-	none
EEM_C750_Values	Function block that converts the EEM_UDT_C750 input structure into individual output parameters.	1	-	none
EEM_C850_Values	Function block that converts the EEM_UDT_C850 input structure into individual output parameters.	1	-	none
EEM_C900_Values	Function block that converts the EEM_UDT_C900 input structure into individual output parameters.	1	-	none
EEM_C950_Values	Function block that converts the EEM_UDT_C950 input structure into individual output parameters.	1	-	none
EEM_CB00_Values	Function block that converts the EEM_UDT_CB00 input structure into individual output parameters.	1	-	none

[*1] with additional module EEM-RS485-MA400 (2901365)

[*2] with additional module EEM-RS485-MA600 (2901367)

[*3] with additional modulel EEM-PB 12-MA600 (2901418)

[*10] Terminated

3.18 FileHandling

Function block	Description	Version	Supported articles	License
FILE_ReadIni	Function block for reading parameters from a file which corresponds to the structure of a classic settings file.	1.00	-	none

3.19 FunctionModules

Function block	Description	Version	Supported articles	License
C_COUNT	Function block for parameterization of the supported terminals in event counting operating mode.	1.00	IB IL CNT (2836337) IB IL CNT-PAC (2861852) IB IL CNT-2MBD (2855813) IB IL CNT-2MBD-PAC (2862071)	none
C_FREQ	Function block for parameterization of the supported terminals in frequency measurement operating mode and cyclic reading out of the measured values.	1.00	IB IL CNT (2836337) IB IL CNT-PAC (2861852) IB IL CNT-2MBD (2855813) IB IL CNT-2MBD-PAC (2862071)	none
C_PULSE	Function block for parameterization of the supported terminals in pulse generator operating mode.	1.00	IB IL CNT (2836337) IB IL CNT-PAC (2861852) IB IL CNT-2MBD (2855813) IB IL CNT-2MBD-PAC (2862071)	none
C_TIME	Function block for parameterization of the supported modules with time measurement operating mode and cyclic reading out of the measured values.	1.00	IB IL CNT (2836337) IB IL CNT-PAC (2861852) IB IL CNT-2MBD (2855813) IB IL CNT-2MBD-PAC (2862071)	none
FUM_IL_DI8_S0_CFG	Function block for parameterizing the channels of the supported modules. Has to be called for every channel.	1.10	IB IL DI 8/S0-PAC (2897020)	none

FUM_IL_DI8_S0_COM	Function block for communication with the supported modules.	1.11	IB IL DI 8/S0-PAC (2897020)	none
FUM_IL_DI8_S0_DATA	Function block for showing the values of a channel. Has to be called for every channel.	1.10	IB IL DI 8/S0-PAC (2897020)	none
FUM_IL_PWM2	Function block for the communication.	2	IB IL PWM/2 (2742612) IB IL PWM/2-PAC (2861632)	none
FUM_IL_PWM2_Para	Function block for parameterization and scaling.	1.00	IB IL PWM/2 (2742612) IB IL PWM/2-PAC (2861632)	none
FUM_IL_PWM2_Data	Function block for diagnosis information.	2	IB IL PWM/2 (2742612) IB IL PWM/2-PAC (2861632)	none
INC_IN	Function block for parameterization and scaling of position values of the supported modules.	1.00	IB IL INC-IN (2819228) IB IL INC-IN-PAC (2819765) IB IL INC-IN-2MBD (2819778) IB IL INC-INC-2MBD-PAC (2819765)	none
INC_DATA	Auxiliary block for assigning structure variables.	1.00	IB IL INC-IN (2819228) IB IL INC-IN-PAC (2819765) IB IL INC-IN-2MBD (2819778) IB IL INC-INC-2MBD-PAC (2819765)	none

INC_PARA	Auxiliary block for assigning structure variables.	1.00	IB IL INC-IN (2819228) IB IL INC-IN-PAC (2819765) IB IL INC-IN-2MBD (2819778) IB IL INC-INC-2MBD-PAC (2819765)	none
IMP_IN	Function block for parameterization and scaling of position values of the supported modules.	1.00	IB IL IMPULSE-IN (2819231) IB IL IMPULSE-IN-PAC (2861768) IB IL IMPULSE-IN-2MBD-PAC (2819804) IB IL IMPULSE-IN-2MBD-PAC (2819804)	none

3.20 HART_Basic

Function block	Description	Version	Supported articles	License
HART_ComAI2	Used for communication with the hardware.	2	IB IL AI2-HART-PAC (2862149)	none
HART_Command1	Displays the measured values received on HART command “1”.	2	–	none
HART_Command2	Displays the measured values received on HART command “2”.	2	–	none
HART_Command3	Displays the measured values received on HART command “3”.	2	–	none
HART_PassThrough	Offers the option of sending additional HART commands to any devices.	2	–	none
HART_DecodeDiag	Decodes the diagnostic code of the function block.	1.00	–	none

3.21 ILCME_MCE

Function block	Description	Version	Supported articles	License
M191_DRV1	Function block (driver block) for connecting the two pulse direction outputs of the Inline controller to the blocks of the Easy Motion library.	1.02	ILC 191 ME/AN (2700074) ILC 191 ME/INC (2700075)	none
M191_DRV2	Function block (driver block) for connecting the two pulse direction outputs of the Inline controller to the blocks of the Easy Motion library.	1.02	ILC 191 ME/AN (2700074) ILC 191 ME/INC (2700075)	none
M191_HomePARA	This function block is used to parameterize the M191_Home function block.	1.00	-	none
M191_Home	This function block is used to perform homing.	1.00	-	none
M191_MoveAbsolute	This function block is used to perform an absolute move.	1.00	-	none
M191_MoveRelative	This function block is used to perform a relative move.	1.00	-	none
M191_MoveVelocity	This function block is used to perform a continuous move.	1.00	-	none
M191_Power	This function block activates or deactivates the other function blocks of the Easy Motion library and enables the pulse direction outputs of the Inline controller.	1.00	-	none
M191_Reset	This function block acknowledges an axis-related error. The axis is set from “ErrorStop” to the “StandStill” state. All axis-related errors are reset. The outputs of the other function blocks remain unaffected.	1.00	-	none
M191_Stop	This function block is used to stop a move using a specified deceleration ramp.	1.00	-	none

3.22 ILCME_Modbus

Function block	Description	Version	Supported articles	License
MB191_485_Para	This auxiliary block is used to parameterize the block for the internal RS-485/RS-422 interface of an ILC 191 ME.	1.00	ILC 191 ME/AN (2700074) ILC 191 ME/INC (2700075)	none
MB191_485_T1	This function block is used to parameterize the internal RS-485/RS-422 interface of an ILC 191 ME.	1.00	ILC 191 ME/AN (2700074) ILC 191 ME/INC (2700075)	none
MB191_RTU_Gateway	This block acts as the gateway and enables communication with Modbus RTU devices.	1.00	-	none
MB191_RTU_FC1FC2	This block covers Modbus function codes 01 and 02.	1.00	-	none
MB191_RTU_FC3FC4	This block covers Modbus function codes 03 and 04.	1.01	-	none
MB191_RTU_FC5	This block covers Modbus function code 05.	1.00	-	none
MB191_RTU_FC6	This block covers Modbus function code 06.	1.00	-	none
MB191_RTU_FC15	This block covers Modbus function code 15.	1.00	-	none
MB191_RTU_FC16	This block covers Modbus function code 16.	1.00	-	none
MB191_RTU_FC23	This block covers Modbus function code 23.	1.00	-	none

3.23 Interbus_PN

Function block	Description	Version	Supported articles	License
IB_AutoSWT_ILC	INTERBUS handling on a PROFINET proxy.	2.00	-	none
IB_AutoSWT_PN	INTERBUS handling on a PROFINET proxy.	2.01	-	none
IB_Control_PN	INTERBUS handling on a PROFINET proxy.	2.01	-	none
IB_DiagReg_PN	INTERBUS handling on a PROFINET proxy.	2.00	-	none
IB_ErrorCode_PN	INTERBUS handling on a PROFINET proxy.	2.01	-	none
IB_ReadConfig	INTERBUS handling on a PROFINET proxy.	2.01	-	none
IB_SendCommand_PN	INTERBUS handling on a PROFINET proxy.	2.00	-	none

3.24 IntrinsicallySafe

Function block	Description	Version	Supported articles	License
IL_EX_IS_AIO4_Com	Function block for communicationnmanagement of the IB IL EX-IS AIO 4/EF-PAC (2869912) module.	1.01	IB IL EX-IS AIO 4/EF-PAC (2869912)	none
IL_EX_IS_AIO4_Data	Function block for parameterization of the IB IL EX-IS AIO 4/EF-PAC (2869912) module.	1.00	IB IL EX-IS AIO 4/EF-PAC (2869912)	none
IL_EX_IS_DIO_4	Function block for parameterization and communication management of the IB IL EX-IS DIO 4/NAM-PAC (2869911) module.	1.01	IB IL EX-IS DIO 4/NAM-PAC (2869911)	none
IL_EX_IS_TEMP_4	Function block for parameterization and communication management of the IB IL EX-IS DIO 4/NAM-PAC (2869911) module. It provides the communication interface to the IB IL EX IS TEMP 4 RTD/TC-PAC (2869913) module.	2	IB IL EX-IS DIO 4/NAM-PAC (2869911) IB IL EX IS TEMP 4 RTD/TC-PAC (2869913)	none

3.25 IOL_Basic

Function block	Description	Version	Supported articles	License
IOL_COM	The block enables the asynchronous communication with the Phoenix Contact IO-Link modules. The function block can be used to write and / or read IO-Link services on the IO-Link-Master or on the IO-Link-Devices	3	AXL E PN IOL8 DI4 M12 6M (2701519) IB IL 24 IOL 4 DI 12-2MBD-PAC (2692733) IB IL 24 IOL 4 DI 12-PAC (2692717) IOL MA8 PN DI8 (1072838)	none

3.26 IP_Com

Function block	Description	Version	Supported articles	License
IPC_Socket	This function block manages the socket firmware function blocks.	5	-	none
IPC_DiagInfo_DE	This function block outputs the last diagnostics of the connection as German text.	2	-	none
IPC_DiagInfo_EN	This function block outputs the last diagnostics of the connection as English text.	2	-	none

3.27 IT_Library

Function block	Description	Version	Supported articles	License
DNS	This function block can be used to request the IP address assigned to a host name from a DNS server.	1.12	-	none
ITL_FTP_FileCopy	This block makes it possible to copy a file between FTP servers.	4	-	none
ITL_FTP_FileRW	This block allows writing to a file on an FTP server or reading from a file on an FTP server.	3	-	none
ITL_FTP_Mngt	Management function block for FTP-protocol.	3	-	none
SetClock	Function block for setting the real time clock.	1.00	-	none
ITL_DHCP_Client	Using the ITL_DHCP_Client function block, a controller is able to request network parameters from the control program via the Dynamic Host Configuration Protocol from a DHCP server.	2.01	-	none
ITL_SNTP_Client	The SNTP_Client block determines the current time of an (S)NTP server via the SNTP protocol and makes this available at its outputs for further processing.	1.02	-	none
ITL_SMTP_Client	The Simple Mail Transfer Protocol (SMTP) is a protocol of the Internet protocol family.	2	-	APPLIC_A*
ITL_SNTP_Diag_Info_EN	In case of an error at the ITL_SNTP_Client, this block shows the diagnostics of the block as a text in English.	1.00	-	none
ITL_SNTP_Diag_Info_DE	In case of an error at the ITL_SNTP_Client, this block shows the diagnostics of the block as a text in German.	1.00	-	none
ITL_HTTP_Get	This block sends a HTTP Get request to a HTTP Server and receives the response.	2	-	none
ITL_HTTP_Post	This block sends a HTTP Get request to a HTTP Server and receives the response.	2	-	none
ITL_HTTP_Decode	This block decodes a byte-array in readable strings. If used with the HTTP_Get or HTTP_Post block, the server response can be decoded directly in plain text.	2	-	none

* If no license is present, the function blocks will work for six hours and then be disabled.

3.28 IT_Security

Function block	Description	Version	Supported articles	License
SEC_AES_CFB_Dec	Function block with AES algorithm for symmetrical decryption – CFB mode.	1.00	-	none
SSEC_AES_CFB_Enc	Function block with AES algorithm for symmetrical encryption – CFB mode.	1.00	-	none
SEC_ AES _Decrypt	Function block with AES algorithm for symmetrical decryption.	1.00	-	none
SEC_ AES _Encrypt	Function block with AES algorithm for symmetrical encryption.	1.00	-	none
SEC_HMAC_SHA1	Function block for encryption with Secure Hash Algorithm (SHA).	1.00	-	none
SEC_SHA1	Function block for encryption with Secure Hash Algorithm (SHA).	1.00	-	none

3.29 Lighting

Function block	Description	Version	Supported articles	License
PLD_PWM	Function block for parameterizing PLD machine lights via the IB IL PWM/2-PAC (2861632) Inline function module.	1.01	IB IL PWM/2-PAC (2861632)	none

3.30 MCE_Library

Function block	Description	Version	Supported articles	License
MCE_Basic				
MCE_Home	MC_Home functionality according to the PLC-OPEN Motion Control Definition	1.01	—	none
MCE_MoveAbsolute	MC_MoveAbsolute functionality according to the PLC-OPEN Motion Control Definition	1.01	—	none
MCE_MoveVelocity	MC_MoveVelocity functionality according to the PLC-OPEN Motion Control Definition	1.01	—	none
MCE_Power	MC_Power functionality according to the PLC-OPEN Motion Control Definition	1.01	—	none
MCE_Reset	MC_Reset functionality according to the PLC-OPEN Motion Control Definition	1.01	—	none
MCE_SetPosition	SetPosition functionality according to the PLC-OPEN Motion Control Definition	1.01	—	none
MCE_Stop	MC_Stop functionality according to the PLC-OPEN Motion Control Definition	1.01	—	none
MCE_TorqueControl	MC_TorqueControl functionality according to the PLC-OPEN Motion Control Definition	1.01	—	none
MCE_Com				
MCE_ETH_C_DIN66019	The block forms the communication interface between the converter-specific driver block and the Ethernet DIN 66019 protocol	1.01	—	none
MCE_C_IB_PCP	The block forms the communication interface between the converter-specific driver block and the PCP protocol	1.01	—	none
MCE_Drive_KEBF5				
MCE_D_KEBF5_DRV	Reading and writing of converter parameters	1.01	KEB-F5 converter	none
MCE_D_KEBF5_FR	Writes the configuration to the converter	1.01	KEB-F5 converter	none
MCE_D_KEBF5_IBI32	Writes the input process data to the axis structure	1.00	KEB-F5 converter	none
MCE_D_KEBF5_IBO32	Writes the axis structure to the output process data	1.00	KEB-F5 converter	none
MCE_D_KEBF5_IBI48	Writes the input process data to the axis structure	1.01	KEB-F5 converter	none
MCE_D_KEBF5_IBO48	Writes the axis structure to the output process data	1.01	KEB-F5 converter	none
MCE_D_KEBF5_IBOP	Block for transmitting parameters to the converter via INTERBUS	1.00	KEB-F5 converter	none
MCE_Util				
MCE_U_DEV_Config	The block is used to transmit configuration data to the communication block	1.00	—	none
MCE_U_DeviceState	The block outputs the status of the converter in the Axis data structure bit by bit	1.00	—	none
MCE_U_PROC_Value	The block outputs the status of the converter in the Axis data structure bit by bit	1.00	—	none

MCE_U_URP	The function block can be used by the user to read parameters from the converter that have not yet been provided	1.00	—	none
MCE_U_UWP	The function block can be used by the user to send parameters to the converters which have not yet been provided	1.00	—	none
MCE_ILCME				
MCE_M191_Config	This block is used to configure the axis. It writes the input parameters into the Axis structure	1	ILC 191 ME/INC (2700075)	none
MCE_M191_DRV	This block is used to control the axis and show diagnosis information	1	ILC 191 ME/INC (2700075)	none
MCE_M191_FWIF1	This block contains the firmware blocks INC_CNT1 and PULSE_CH1. This block describes the onboard PWM output channel 1 and the onboard INC input channel 1 is queried.	1	ILC 191 ME/INC (2700075)	none
MCE_M191_FWIF2	This block contains the firmware blocks INC_CNT2 and PULSE_CH2. This block describes the onboard PWM output channel 2 and the onboard INC input channel 2 is queried.	1	ILC 191 ME/INC (2700075)	none
MCE_DiagInfo_DE	This block converts the diagnostic information into a text in German	1	ILC 191 ME/INC (2700075)	none
MCE_DiagInfo_EN	This block converts the diagnostic information into a text in English	1	ILC 191 ME/INC (2700075)	none

3.31 Modbus_RTU

Function block	Description	Version	Supported articles	License
MB_RTU_FC1	This function block reads the status of discrete outputs from a Modbus slave.	7	-	none
MB_RTU_FC2	This function block reads discrete inputs from a Modbus slave.	8	-	none
MB_RTU_FC3	This function block reads holding registers from a Modbus slave.	7	-	none
MB_RTU_FC4	This function block reads input registers from a Modbus slave.	7	-	none
MB_RTU_FC5	This function block writes a single output bit of a Modbus slave.	7	-	none
MB_RTU_FC6	This function block writes a single holding register of a Modbus slave.	7	-	none
MB_RTU_FC15	This function block writes multiple output bits of a Modbus slave.	7	-	none
MB_RTU_FC16	This function block writes multiple holding registers of a Modbus slave.	7	-	none
MB_RTU_FC23	This function block writes or reads multiple holding registers of a Modbus slave.	7	-	none
MB_RTU_DiagInfo_DE	This optional function block displays diagnostic messages of the Modbus master as clear text in German.	3	-	none
MB_RTU_DiagInfo_EN	This optional function block displays diagnostic messages of the Modbus master as clear text in English.	3	-	none
MB_AXL_SE_RS485_Master	This block runs the sending operations via the AXL SE RS485 (1088128) module.	2	AXL SE RS485 (1088128)	none
MB_AXL_SE_RS485_Slave	This block runs the sending operations via the AXL SE RS485 (1088128) module.	1	AXL SE RS485 (1088128)	none
MB_AXL_F_RSUNI_Master	This block runs the sending operations via the AXL F RS UNI 1H (2688666) module.	2	AXL F RS UNI 1H (2688666)	none
MB_AXL_F_RSUNI_Slave	This block runs the sending operations via the AXL F RS UNI 1H (2688666) module.	1	AXL F RS UNI 1H (2688666)	none
MB_IL_232P_Master	This function block is used to implement a Modbus Master for the specified module type.	6	IB IL RS 232-PRO-PAC (2878722)	none
MB_IL_232P_Slave	This function block is used to implement a Modbus Slave for the specified module type.	4	IB IL RS 232-PRO-PAC (2878722)	none
MB_IL_232E_Master	This function block is used to implement a Modbus Master for the specified module type.	4	IB IL RS 232-ECO (2702141)	none

MB_IL_232E_Slave	This function block is used to implement a Modbus Slave for the specified module type.	3	IB IL RS 232-ECO (2702141)	none
MB_IL_485P_Master	This function block is used to implement a Modbus Master for the specified module type.	6	IB IL RS 485/422-PRO-PAC (2863627)	none
MB_IL_485P_Slave	This function block is used to implement a Modbus Slave for the specified module type.	4	IB IL RS 485/422-PRO-PAC (2863627)	none
MB_IL_485E_Master	This function block is used to implement a Modbus Master for the specified module type.	7	IB IL RS 485-ECO (2702795)	none
MB_IL_485E_Slave	This function block is used to implement a Modbus Slave for the specified module type.	5	IB IL RS 485-ECO (2702795)	none
MB_IL_UNIxx_Master	This function block is used to implement a Modbus Master for the specified module type.	7	IB IL RS UNI-PAC (2700893)	none
MB_IL_UNIxx_Slave	This function block is used to implement a Modbus Slave for the specified module type.	5	IB IL RS UNI-PAC (2700893)	none

3.32 Modbus_TCP

Function block	Description	Version	Supported articles	License
MB_TCP_Client	The function block enables communication as client with Modbus TCP devices (server).	3	-	none
MB_TCP_Server	The function block enables communication as a server with a TCP client.	4	-	none
MB_TCP_DiagInfo_EN	This optional function block displays diagnostic messages of the Modbus client as plain text in English.	1	-	none
MB_TCP_FC1	This function code is used to read from 1 to 2000 contiguous status of coils in a remote device.	1	-	none
MB_TCP_FC2	This function code is used to read from 1 to 2000 contiguous status of discrete inputs in a remote device.	1	-	none
MB_TCP_FC3	This function code is used to read from 1 to 125 contiguous holding registers in a remote device.	1	-	none
MB_TCP_FC4	This function code is used to read from 1 to 125 contiguous input registers in a remote device.	1	-	none
MB_TCP_FC5	This function code is used to write a single output to either ON or OFF in a remote device.	1	-	none
MB_TCP_FC6	This function code is used to write a single holding register in a remote device.	1	-	none
MB_TCP_FC15	This function code is used to force each coil in a sequence of coils to either ON or OFF in a remote device.	2	-	none
MB_TCP_FC16	This function code is used to write a block of contiguous registers (1 to 123 registers) in a remote device.	2	-	none
MB_TCP_FC23	This function code performs a combination of one read operation and one write operation in a single MODBUS transaction.	2	-	none

3.33 OperatorPanel

Function block	Description	Version	Supported articles	License
OP_RS232_Control	Function block for setting baud rate for communication between a controller and operator panel (OP) and establishing a connection to the OP.	1.00	NLC-OP2-LCD-076-4X20 (2701945)	none
OP_Write_Strings	Function block for writing up to 8 lines of 20 characters to the screen.	1.00	"	none
OP_Bargraph	Function block for displaying one or two bar graphs.	1.00	"	none
OP_Read_Real	Function block for entering a real value.	1.00	"	none
OP_Read_Time	Function block for entering time values in one of two formats.	1.00	"	none
OP_Backlight	Function block for controlling the illumination in green, red or blue of the display.	1.00	"	none

3.34 PDPI_Basic

Function block	Description	Version	Supported articles	License
PDPI_B_Controller	The PDPI_Controller can be operated as 2-point, 3-point, motor step or continuous controller. It outputs both a continuous, analog signal and a pulse-width-modulated, digital manipulated variable signal separately for heating and cooling (inverse and direct control).	2.11	—	PDPI-BASIC license
PDPI_B_Para	Function block for parameterizing the controller.	2.00	—	none
PDPI_B_Config	Function block for configuration of the controller.	2.00	—	none
PDPI_B_TunePara	PID parameters of the controller.	2.00	—	none
PDPI_B_OutTunePara	PID parameters of the controller from self-optimization.	2.00	—	none
PDPI_B_State	Controller status.	2.00	—	none
PDPI_B_Alarm	Function block for representation of alarms.	2.00	—	none
PDPI_B_Scale	Function for linear scaling of a continuous manipulated variable.	2.00	—	none

3.35 PDPI_Pro

Function block	Description	Version	Supported articles	License
PDPI_P_Controller	The PDPI_Controller can be operated as 2-point, 3-point, motor step or continuous controller. It outputs both a continuous, analog signal and a pulse-width-modulated, digital manipulated variable signal separately for heating and cooling (inverse and direct control).	2.11	–	PDPI PRO license
PDPI_P_Para	Parameterization of the controller.	2.00	–	none
PDPI_P_Config	Function block for configuration of the controller.	2.00	–	none
PDPI_P_HotrunPara	Hot channel controller status.	2.00	–	none
PDPI_P_Limit	Setting of the limit value monitoring.	2.00	–	none
PDPI_P_TunePara	PID parameters of the controller.	2.00	–	none
PDPI_P_OutTunePara	PID parameters of the controller from self-optimization.	2.00	–	none
PDPI_P_Scale	Scaling function	3	–	none
PDPI_P_State	Controller status.	2.00	–	none
PDPI_P_StateHotrun	Hot channel controller status.	2.00	–	none
PDPI_P_Alarm	Function block for representation of alarms.	2.00	–	none
PDPI_P_Cascade	The FB PDPI_Cascade_V2 is the link between the output of the master controller (Ctrl_1_Out) and the set point input of the slave controller (Ctrl_2_SetPoint).	2.00	–	none
PDPI_P_Switch	If a closed-loop control circuit only has one actuator but two sensors, whereby one of the two sensors should be used according to the operating state, one of the corresponding closed-loop control circuits can be activated via a switching controller. This is possible using the FB PDPI_Switch_V2.	2.00	–	none
PDPI_P_Phasenheber	A phase increase can lead to improvement in some difficult closed-loop control circuits that are prone to instability or strong over or undershooting.	2.00	–	none

3.36 PN_Dev_Diag

Function block	Description	Version	Supported articles	License
PNDD_Status	Function block for cyclically reading diagnostic status and diagnostic messages from PNIO devices.	3	All PNIO devices	none
PNDD_IL_Diag	The function block checks the status of Inline modules using its node ID. Diagnostic messages of a PROFIsafe module can be acknowledged using this function block.	3	Inline modules	none
PNDD_AXL_Diag	The function block checks the status of Axioline safety modules using its node ID. Diagnostic messages of a PROFIsafe module can be acknowledged using this function block.	3	AXL F PSDI8/4 1F (2701559) AXL F PSDO8/3 1F (2701560)	none
PNFD_IL_Diag	The function block checks the status of safety Inline I/O modules using its PROFIsafe Address. Diagnostic messages of a PROFIsafe module can be acknowledged using this function block.	3	Safety-Inline modules	none
PNFD_AXL_Diag	The function block checks the status of safety Axioline I/O modules using its PROFIsafe Address. Diagnostic messages of a PROFIsafe module can be acknowledged using this function block.	3	AXL F PSDI8/4 1F (2701559) AXL F PSDO8/3 1F (2701560)	none

3.37 Positioning

Function block	Description	Version	Supported articles	License
IL_INC	Function block for processing the initialization, parameterization, diagnostics and control of the IB IL INC-PAC (2861849) positioning terminal.	1.00	IB IL INC (2836324) IB IL INC-PAC (2861849)	none
IL_SSI	Function block for initialization, parameterization, control, and diagnostics of the IB IL SSI (2836340) positioning terminal.	1.00	IB IL SSI (2836340) IB IL SSI-PAC (2861865) IB IL SSI-2MBD (2855729)	none
IL_SSI_IN	Function block for parameterizing the IB IL SSI-IN-PAC (2819574) terminal	1.04	IB IL SSI-IN (2819309) IB IL SSI-IN-PAC (2819574)	none
RAMP_GENERATOR	This function block makes it possible to achieve any location through a ramping process.	1.00	—	none

3.38 PowerMeasurement

Function block	Description	Version	Supported articles	License
PM_3P_N_EF	Function block for parameterization and communication	1	IB IL PM 3P/N/EF-PAC (27009650)	none
PM_BasicValue	Function block for representing the basic measured values.	1	"	none
PM_ConfDataSample	Function block for defining a data record of up to 11 measured values. These values are read via process data.	1	"	none
PM_ConfigTable	Function block for reading the configuration which applies in the module.	1	"	none
PM_Configuration	Function block for configuration	1	"	none
PM_Diag	Function block for diagnostics.	1	"	none
PM_Energy	Function block for representing the energy measured values.	1	"	none
PM_ExtendedValue	Function block for representing the extended measured values.	1	"	none
PM_H_Current	Function block for representing the 1st to 31st harmonics of the phase current.	1	"	none
PM_HD_Voltage	Function block for representing the phase voltage and the proportions of the 2nd to 31st harmonics of the phase voltage based on the fundamental wave.	1	"	none
PM_IntervalPower	Function block for representing the power profiles.	1	"	none
PM_MaxBasicValue	Function block for representing the maximum basic measured values.	1	"	none
PM_MaxExtendedVal	Function block for representing the maximum extended measured values.	1	"	none
PM_MaxH_Current	Function block for representing the maximum values of the 1st to 31st harmonics of the phase current.	1	"	none
PM_MaxHD_Voltage	Function block for representing the maximum values of the phase voltage and the proportions of the 2nd to 31st harmonics of the phase voltage based on the fundamental wave.	1	"	none
PM_MaxPower	Function block for representing the maximum power measured values.	1	"	none
PM_MaxReactivePower	Function block for representing the maximum reactive power measured values and minimum cos(phi).	1	"	none
PM_MaxTHD	Function block for representing the maximum values of distortion for voltage and current.	1	"	none
PM_Power	Function block for representing the power measured values.	1	"	none
PM_ReactivePower	Function block for representing the reactive power measured values and cos(phi).	1	"	none

PM_SamplingValue	Function block for representing the sampling values for voltage, current, and real power in the sampling values operating mode.	1	"	none
PM_Select_PCP	Function block for selecting the measured values to be read via PCP in the basic measured values operating mode.	1	"	none
PM_Select_PD	Function block for selecting the measured values to be read via process data in the basic measured values operating mode.	1	"	none
PM_SelectSamp_PD	Function block for selecting the measured values to be read via process data in the sampling values operating mode.	1	"	none
PM_SetEnergyValue	Function block for reinitializing a power meter	1	"	none
PM_Sync1Ph	Function block for representing the measured values for network synchronization, only in the synchronization operating mode.	1	"	none
PM_THD	Function block for representing the distortion for voltage and current.	1	"	none

3.39 PROFIdrive

Function block	Description	Version	Supported articles	License
PDRV_ST1	The function block prepares the process data for status word 1 and the actual value in the application program.	1.00	—	none
PDRV_ST1_PD_IN	The function block converts the incoming process array of the device into individual parameters.	1.00	—	none
PDRV_ST1_PD_OUT	The function block converts the individual parameters, such as control word and set point value, into the outgoing process array for the device.	1.00	—	none
PDRV_ST2	The block prepares the process data for status word 1, status word 2, and the actual value in the application program.	1.00	—	none
PDRV_ST2_PD_IN	The function block converts the incoming process array of the device into individual parameters, such as status word and actual value	1.00	—	none
PDRV_ST2_PD_OUT	The function block converts the individual parameters, such as control word and set point value, into the outgoing process array for the device.	1.00	—	none
PDRV_Param_Read	The function block is designed for read access to parameters, as defined in the PROFIdrive profile.	1.00	—	none
PDRV_Param_Change	The function block is designed for write access to parameters, as covered in the PROFIdrive profile.	1.00	—	none
PDRV_CheckChgResp	The PDRV_CheckChgResp block is suitable for evaluating the response to a Change Parameter Request.	1.00	—	none
PDRV_Param_Diag	The PDRV_Param_Diag block converts the information regarding the negative feedback to a request into a text message.	1.00	—	none
PDRV_DiagInfo	This function block converts the diagnostic code of the PDRV_Param_Read, PDRV_Param_Change, and PDRV_Value2 function blocks into a text message.	1.00	—	none
PDRV_Value2ArrByte	This function block extracts the array of byte out of the read parameter response.	1.00	—	none
PDRV_Value2ArrUint	This function block extracts the array of unsigned interger out of the read parameter response.	1.00	—	none
PDRV_Value2ArrWord	This function block extracts the array of word out of the read parameter response.	1.00	—	none
PDRV_Value2Descr	This function block extracts the value as a description of the read parameter response.	1.00	—	none
PDRV_Value2Octet	This function block extracts the value as octet string (byte array) out of the read parameter response.	1.00	—	none

PDRV_Value2Uint	This function block extracts the value as unsigned integer out of the read parameter response.	1.00	—	none
PDRV_Value2Word	This function block extracts the value as word out of the read parameter response.	1.00	—	none
PDRV_ArrByte2Value	This function block is used for preparing an arry of byte for the change parameter request.	1.00	—	none
PDRV_ArrUint2Value	This function block is used for preparing an arry of unsigned integer for the change parameter request.	1.00	—	none
PDRV_ArrWord2Value	This function block is used for preparing an arry of word for the change parameter request.	1.00	—	none
PDRV_Descr2Value	This function block is used for preparing a string for the change parameter request.	1.00	—	none
PDRV_Octet2Value	This function block is used for preparing an octet string for the change parameter request.	1.00	—	none
PDRV_Uint2Value	This function block is used for preparing a value of the data type uint 16 for the change parameter request.	1.00	—	none
PDRV_Word2Value	This function block is used for preparing a word for the change parameter request.	1.00	—	none
PDRV_PPO6_Control	Converts the control data to output process data	2	—	none
PDRV_PPO6_Status	Converts input process data to status data	2	—	none

3.40 RadiolineBasic

Function block	Description	Version	Supported articles	License
RAD_Search_IO	Detecting all the I/O modules in a Radioline network and entering the information in a structure.	2	RAD-2400-IFS (2901541) RAD-868-IFS (2904909) RAD-900-IFS (2901540) RAD-RS485-IFS (2702184)	none
RAD_DIAG	Function block for reading out diagnostic information.	2	RAD-2400-IFS (2901541) RAD-868-IFS (2904909) RAD-900-IFS (2901540) RAD-RS485-IFS (2702184)	none
RAD_DI4	Driver block for RAD-DI4-IFS.	2	RAD-DI4-IFS (2901535)	none
RAD_DI8_STAT	Driver block for RAD-DI8-IFS.	2	RAD-DI8-IFS (2901539)	none
RAD_DI8_CNT	Driver block for RAD-DI8-IFS.	2	RAD-DI8-IFS (2901539)	none
RAD_DOR4	Driver block for RAD-DOR4-IFS.	2	RAD-DOR4-IFS (2901536)	none
RAD_DO8	Driver block for RAD-DO8-IFS.	2	RAD-DO8-IFS (2902811)	none
RAD_AI4	Driver block for RAD-AI4-IFS.	2	RAD-AI4-IFS (2901537)	none
RAD_AO4	Driver block for RAD-AO4-IFS.	2	RAD-AO4-IFS (2901538)	none
RAD_PT100_4	Driver block for RAD-PT100-4-IFS.	2	RAD-PT100-4-IFS (2904035)	none
RAD_DAIO6	Driver block for RAD-DAIO6-IFS.	2	RAD-DAIO6-IFS (2901533)	none
RAD_NAM4	Driver block for RAD-NAM4-IFS.	1	RAD-NAM4-IFS (2316275)	none
RAD_AI4_U	Driver block for RAD-AI4-IFS.	1	RAD-AI4-IFS (2901537)	none

3.41 RDNCY

Function block	Description	Version	Supported articles	License
RED_CTD	This counter function block counts down.	1.01	All PLCs	none
RED_CTU	This counter function block counts up.	1.01	All PLCs	none
RED_CTUD	This counter function block counts up or down.	1.01	All PLCs	none
RED_IP_CONNECT	This IP Connect block establishes a communication connection.	1.00	RFC 460R PN 3TX (2700784)	none
RED_RS	This bistable function block implements the dominant resetting of the Q1 output.	2.00	All PLCs	none
RED_SR	This bistable function block implements the dominant setting of the Q1 output.	2.00	All PLCs	none
RED_TOF	This timer implements a switch-off delay.	2.00	All PLCs	none
RED_TOF_R	This timer implements a switch-off delay.	1.00	RFC 460R PN 3TX (2700784)	none
RED_TON	This timer implements a switch-on delay.	2.00	All PLCs	none
RED_TON_R	This timer implements a switch-on delay.	1.00	RFC 460R PN 3TX (2700784)	none
RED_TP	This timer generates a pulse.	3	All PLCs	none

3.42 Redundancy_ASR

Function block	Description	Version	Supported articles	License
RED_ContrRed_3xx	This function block determines the role of master or slave on the primary and backup controllers.	1.00	-	none
RED_ContrRed_4xx	This function block determines the role of master or slave on the primary and backup controllers.	1.00	-	none

3.43 SBT_Technology

Function block	Description	Version	Supported articles	License
SBT_IBS_Download	Function block for downloading the configuration and parameter data record of a SAFECONF security program to an LPSDO in an INTERBUS system.	1.04	IB IL 24 LPSDO 8-PAC (2916024)	none
SBT_Operate	Function block for copying process data between the LPSDO and the connected PSDI of the corresponding SafetyBridge island.	1.03	IB IL 24 LPSDO 8-PAC (2916024)	none
SBT_PN_Download	Function block for downloading the configuration and parameter data record of a SAFECONF security program to an LPSDO in a PROFINET system.	1.04	IB IL 24 LPSDO 8-PAC (2916024)	none
SBT_V2_BinFile	Function block for reading in a SAFECONF project.	1.00	IB IL 24 LPSDO 8 V2-PAC (2700606)	none
SBT_V2_IBS_Restart	Function block for restarting the INTERBUS system.	1.00	IB IL 24 LPSDO 8 V2-PAC (2700606)	none
SBT_V2_Operate3	Function block for sending the LPSDO project and exchanging data between the LPSDO and devices.	1.01	IB IL 24 LPSDO 8 V2-PAC (2700606)	none
SBT_V2_Operate5	Function block for sending the LPSDO project and exchanging data between the LPSDO and devices.	1.01	IB IL 24 LPSDO 8 V2-PAC (2700606)	none
SBT_V2_ProjHeader	Function block for itemizing the project data.	1.00	IB IL 24 LPSDO 8 V2-PAC (2700606)	none
SBT_V2_TransTime	Function block for displaying the data transmission time.	1.00	IB IL 24 LPSDO 8 V2-PAC (2700606)	none

3.44 SBT_V3

Function block	Description	Version	Supported articles	License
SBT_V3_Operate	• Operates a safety island • Downloads a SAFECONF project • Monitors the input and output modules	4	AXL F LPSDO8/3 1F (2702171) IB IL 24 LPSDO 8 V3-PAC (2701625)	none
SBT_V3_CrossComm	The function block enables cross communication between the secure islands	4	AXL F LPSDO8/3 1F (2702171) IB IL 24 LPSDO 8 V3-PAC (2701625)	none
SBT_V3_DataExch	If two islands are connected with different controllers and cross-communication is needed, then the function block is used for data transmission between the master and the slave island.	4	AXL F LPSDO8/3 1F (2702171) IB IL 24 LPSDO 8 V3-PAC (2701625)	none
SBT_V3_InBuff	The block links up the input process data of the devices (modules).	4	AXL F LPSDO8/3 1F (2702171) IB IL 24 LPSDO 8 V3-PAC (2701625)	none
SBT_V3_OutBuff	The block links up the output process data of the devices (modules).	4	AXL F LPSDO8/3 1F (2702171) IB IL 24 LPSDO 8 V3-PAC (2701625)	none
SBT_V3_BinFile	A SAFECONF project can be created as a BIN file, which is saved on the FTP server of the controller. The function block reads the project data and provides the SBT_V3_Operate function block with the data.	4	AXL F LPSDO8/3 1F (2702171) IB IL 24 LPSDO 8 V3-PAC (2701625)	none

SBT_V3_DiagCode	The block shows the contents of the arr_wSBTdiagCode array. The diagnostics descriptions are in the user manual for the corresponding module.	4	AXL F LPSDO8/3 1F (2702171) IB IL 24 LPSDO 8 V3-PAC (2701625)	none
SBT_V3_DiagInfo	The block converts the diagnostic codes according to the user manuals and outputs these as text.	1	All SBT devices	none
SBT_V3_ProjHeader	This block represents the project data from the file (BIN file/POU), and from the downloaded project on the LPSDO, as output parameters.	4	AXL F LPSDO8/3 1F (2702171) IB IL 24 LPSDO 8 V3-PAC (2701625)	none
SBT_V3_TransTime	This block indicates the data transmission time for each satellite (module).	4	AXL F LPSDO8/3 1F (2702171) IB IL 24 LPSDO 8 V3-PAC (2701625)	none
SBT_V3_Tracer	If a communication error occurs, the block represents the last four telegrams between the AXL F LPSDO8/3 1F (2702171) and the modules.	2	AXL F LPSDO8/3 1F (2702171) IB IL 24 LPSDO 8 V3-PAC (2701625)	none
SBT_V3_IP_CycCom	The block allows cross communication from LPSDO modules to different controllers.	2	-	none
SBT_V3_IBS_Restart	The block is used for starting up the INTERBUS in the event of problems when starting up the ILCxxx controller.	2	IB IL 24 LPSDO 8 V3-PAC (2701625)	none
SBT_V3_PS_TO_SBT	The function block allows cross communication between LPSDO and PROFISAFE Controller.	1	AXL F LPSDO8/3 1F (2702171) IB IL 24 LPSDO 8 V3-PAC (2701625)	none

3.45 SimpleLogger

Function block	Description	Version	Supported articles	License
SLO_SimpleLogger	Function block for logging a maximum of nine strings each cycle.	2	-	none

3.46 SNMP

Function block	Description	Version	Supported articles	License
SNMP_Agent	The SNMP Agent block represents a server that enables access to the information of the control program using the Simple Network Management Protocol (SNMP).	3	—	None
SNMP_Client	The SNMP Client block is used for communication with a remote SNMP agent.	3	—	None
SNMP_TRecv	The Trap Receiver block listens to incoming traps, checks them, and transmits the data to the control program. The block supports traps according to protocol version 2c and can process a lot of additional information, the variable bindings, from the received trap.	3	—	None
SNMP_TSend	The Trap Sender block allows transmission of alarm messages. The block packs the transmitted data into a SNMP trap according to protocol version 2c and sends this trap. A lot of additional information, the variable bindings, can be sent with the trap.	3	—	None
SNMP1_TRecv	The Trap Receiver block listens to incoming messages, checks them, and transmits the data to the control program. Traps based on SNMP v1 that contain multiple variable bindings are supported.	2	—	None

3.47 SNMP3

Function block	Description	Version	Supported articles	License
SNMP3_Agent	The function block SNMP3_Agent represents a server that enables access to control program information over the Simple Network Management Protocol (SNMP).	2.00	—	none
SNMP3_TRecv	The function block SNMP3_TRecv listens for incoming messages over the Simple Network Management Protocol (SNMP), checks them and forwards the data to the control program.	2.00	—	none
SNMP3_TSend	The function block SNMP3_TSend enables transmitting of alarm messages by the control program. Data forwarded to the block is converted to a version 2c SNMP trap whereby several variables in the form of Variable Bindings can be added to it.	2.00	—	none

3.48 SYS_PLC

Function block	Description	Version	Supported articles	License
SYS_BufToString80	This function block converts a byte array containing ASCII characters into a STRING value according to IEC 61131 during one PLC cycle.	1.00	-	none
SYS_CheckIPv4String	This function block allows user to check IP addresses version 4.	1.00	-	none
SYS_CheckLicense	This function block checks the installed memory card license against the license requirements for the application according to the Phoenix Contact PLC licensing specification.	1.00	-	none
SYS_CheckVarName	This function block checks whether a given string is a valid variable name according to IEC 61131-3.	1.00	-	none
SYS_CheckVarValue	This function block checks whether a given string is a valid value according to IEC 61131-3. If the value is valid, the function block returns a datatype number of the detected type.	1.00	-	none
SYS_CTrig	This function block detects rising and falling edges of the input signal.	1.00	-	none
SYS_CycleTime	This function block calculates the time between two function block calls of one instance with a resolution of 1 ms. This can be used to detect task cycle times.	1.00	-	none
SYS_DtDayOfMonth	This function extracts the current day of the month from the Unix time. Unix format: time expired since 1.1.1970, 00:00 in seconds.	1.00	-	none
SYS_DtDayOfUnixUtc	This function extracts the current day from Unix time. UNIX format: time expired since 1.1.1970 00:00 in seconds.	1.00	-	none
SYS_DtDayOfWeek	This function resolves the number of the day within the week and returns a value according to ISO 8601.	1.00	-	none
SYS_DtDayOfYear	This function extracts the current day of the year from the Unix time. Unix format: time expired since 1.1.1970, 00:00 in seconds.	1.00	-	none
SYS_DtGetMsFromPlcTick	This function blocks returns the elapsed time in milliseconds since last call.	1.00	-	none
SYS_DtEmulatedRtc	This function block is used to calculate a time stamp included milliseconds.	1.00	-	none
SYS_DtHourOfDay	This function extracts the current hour of the day from the Unix time. Unix format: time expired since 1.1.1970, 00:00 in seconds.	1.00	-	none

SYS_DtLeapYear	This function evaluates a given time in Unix time format and determines whether the current year is a leap year.	1.00	-	none
SYS_DtMinuteOfHour	This function extracts the current month of the year from the Unix time. Unix format: time expired since 1.1.1970, 00:00 in seconds.	1.00	-	none
SYS_DtMonthOfYear	This function extracts the current minute of an hour from the Unix time. Unix format: time expired since 1.1.1970, 00:00 in seconds.	1.00	-	none
SYS_DtSecondOfMinute	This function extracts the current second of a minute from the Unix time. Unix format: time expired since 1.1.1970, 00:00 in seconds.	1.00	-	none
SYS_DtSummerTimeEurope	This function returns TRUE, if daylight saving time is active.	1.00	-	none
SYS_DtUnixUtcToISO8601	This function block converts a time stamp given in Unix time format to time stamp according ISO 8601 as STRING.	1.00	-	none
SYS_DtUnixUtcToLocal	This function converts the UTC time into the local time.	1.00	-	none
SYS_DtUtcToUnixUtc	This function converts the time given by input parameters: Year, Month, Day, Hour, Minute and Second into Unix format time.	1.00	-	none
SYS_DTWeekOfYear	This function extracts current week of year from Unix timestamp. UNIX time format: time expired since 1.1.1970 00:00 in seconds.	1.00	-	none
SYS_DtYearOfUnixUtc	This function extracts the current year from the Unix time. Unix format: time expired since 1.1.1970, 00:00 in seconds.	1.00	-	none
SYS_EventLogData	The SYS_EventLogData function block writes a log entry in the internal buffer of the event logger function block.	1.00	-	none
SYS_EventLogHeader	The SYS_EventLogHeader function block configures the file header for an event log file.	1.00	-	none
SYS_EventLogWriter	The SYS_EventLogWriter function block writes the internal buffer of the event logger in a file.	1.00	-	none
SYS_FileCopy	This function block is creating a copy of an existing file on the local file system of the PLC.	1.00	-	none
SYS_FileGateway	This function block is a universal file system interface.	1.00	-	none
SYS_FileRemove	This function block deletes a file on the programmable logic controller (PLC).	1.00	-	none
SYS_FileSize	This function block is used to read the file size in bytes. Please note that if the source file does not exist an empty file is create.	1.00	-	none
SYS_FilesOfDir	This function block is used to read all file names within a directory.	1.00	-	none

SYS_FolderMngt	This function block is used to create a folder structure on the file system of the PLC including FTP access rights.	1.00	-	none
SYS_GetDatatypeID	This function block returns the internal data type ID for non-generic IEC 61131 data types.	1.00	-	none
SYS_GetVar	This function block returns an IEC 61131-compliant string containing the variable value of a named variable.	1.00	-	none
SYS_IEEE_ToLReal	This function block converts IEEE float variables into IEC 61131 LREAL variables.	1.00	-	none
SYS_IEEE_ToReal	This function block converts IEEE float variables into IEC 61131 REAL variables.	1.00	-	none
SYS_IPv4ToString	This function block allows user to convert IP addresses version 4 given as DWORD to STRING.	1.00	-	none
SYS_LRealTo_IEEE	This function block converts IEC 61131 LREAL variables into IEEE float variables.	1.00	-	none
SYS_PlcCtrl	Function block for the parameterization of PLCs.	1.00	-	none
SYS_PlcExternalCard	This function block allows user to enable external card.	1.00	-	none
SYS_PlcFlahCardInfo	This function block allows user to read information about flash card.	1.00	-	none
SYS_PlcFormatRamDisk	Function block for formatting the RAM disk (virtual HDD within the RAM) on programmable logic controllers (PLCs).	1.00	-	none
SYS_PlcFtp	Function block for activating/deactivating the FTP server on PLCs.	1.00	-	none
SYS_PlcFwUpdate	Function block for performing firmware updates on PLCs.	1.00	-	none
SYS_PlcGetNetloadSet	This function block allows user to read netload limiter settings of available network adapter(s) on the programmable logic controller (PLC).	1.00	-	none
SYS_PlcGetNetworkSet	This function block allows user to get current network settings such as: Number of network adapters, IP address (IPv4/IPv6), Subnet mask (IPv4/IPv6), Gateway IP address (IPv4/IPv6), MAC address for desired network adapter.	1.00	-	none
SYS_PlcGetPNctrlStat	This function block allows user to read status of Profinet controller on programmable logic controller (PLC).	1.00	-	none
SYS_PlcGetPNdevStat	This function block allows user to read state of Profinet device on programmable logic controller (PLC).	1.00	-	none
SYS_PlcGetRTC	This function block allows user to read real time clock (RTC) on programmable logic controller (PLC).	1.00	-	none
SYS_PlcGetSntpStat	This function block reads current configuration of SNTP client on the programmable logic controller (PLC).	1.00	-	none

SYS_PlcGetSystemInfo	This function block allows user to read system information of programmable logical controller (PLC).	1.00	-	none
SYS_HmiService	This function block allows user to sendfirmware services via WEB HMI using special codes onto programmable logic controller (PLC).	1.00	-	none
SYS_PlcHttp	Function block for activating/deactivating the HTTP server on PLCs.	1.00	-	none
SYS_PlcLogNetloadSet	This function block allows user to activate logging of netload limiter settings for available network adapter(s) on the programmable logic controller (PLC).	1.00	-	none
SYS_PlcLogNetloadStat	This function block allows user to activate logging of netload limiter statistics for available network adapter(s) on the programmable logic controller (PLC).	1.00	-	none
SYS_PlcNetloadLimiter	Function block is used for limiting the network load and consequently avoiding an overload on programmable logical controllers (PLCs).	1.00	-	none
SYS_PlcNetloadStat	Function block for reading of network load statistics of available network adapters on programmable logical controllers (PLC).	1.00	-	none
SYS_PlcNetworkSet	Function block for changing a saved network setting on programmable logic controllers (PLC).	1.00	-	none
SYS_PlcProfiNetCtrl	This function block allows user to disable/enable ProfiNet controller instance.	1.00	-	none
SYS_PlcProfiNetDevice	This function block allows user to disable/enable ProfiNet device instance.	1.00	-	none
SYS_PlcRedResetError	This function block allows user to reset last error on redundant programmable logic controller (PLC). This function is only available on RFC460R.	1.00	-	none
SYS_PlcReset	Function block for resetting programmable logic controllers (PLC).	1.00	-	none
SYS_PlcSetPnDevNetAd	This function block allows user to define, which network interface of the programmable logic controller (PLC) is used by its PROFINET IO device.	1.00	-	none
SYS_PlcSetPnDevNetAd	This function block allows user to define, which network interface of the programmable logic controller (PLC) is used by its PROFINET IO device.	1.00	-	none
SYS_PlcSetNetPNdev	This function block sets the network interface for the PROFINET IO device on programmable logic controller (PLC), provided PLC supports multiple network interfaces.	1.00	-	none
SYS_PlcSetRTC	This function block allows user to set real time clock on programmable logical controller (PLC).	1.00	-	none

SYS_PlcSnmp	Function block for activating/deactivating a SNMP server on programmable logic controllers (PLC).	1.00	-	none
SYS_PlcSntpClient	Function block for parameterization and activation of the SNTP client on programmable logic controllers (PLC).	1.00	-	none
SYS_PlcSwitchRole	Function block for switching the redundancy role on programmable logic controllers (PLC).	1.00	-	none
SYS_RealTo_IEEE	This function block converts IEC 61131 REAL variables into IEEE float variables.	1.00	-	none
SYS_SetVar	This function block writes an IEC 61131-compliant string containing the variable value to a named variable.	1.00	-	none
SYS_SPlcGetCycleTime	This function block allows the user to read the cycle time on a safety programmable logic controller (SPLC). This function is only available for RFC470S controllers.	1.00	-	none
SYS_SPlcSetCycleTime	This function block allows the user to set the cycle time on a safety programmable logical controller (SPLC). This function is only available for RFC 470S controllers.	1.00	-	none
SYS_StringToIPv4	This function block allows user to convert IP addresses version 4 given as STRING to DWORD.	1.00	-	none
SYS_StringToLower	This function block converts upper case letters of a string to lower case letters.	1.00	-	none
SYS_StringToTime	This function block converts an IEC 61131 time string value to the TIME data type of the runtime environment.	1.00	-	none
SYS_StringToUpper	This function block converts lower case letters of a string to upper case letters.	1.00	-	none
SYS_VerToNum	Function block for extracting and copying of version numbers of a version string into a double word variable.	1.00	-	none

3.49 TempConversion

Function block	Description	Version	Supported articles	License
IL_TEMP_2_xxx	Function block for the parameterization and operation of the IB IL TEMP 2 RTD and IB IL TEMP 2 UTH Inline modules.	1.10	IB IL TEMP 2 RTD-PAC (2726308) IB IL TEMP 2 UTH-PAC (2861386)	none
IL_TEMP_2_xxx_Para	Function block for the parameterization of the IB IL TEMP 2 RTD and IB IL TEMP 2 UTH Inline modules.	1.00	IB IL TEMP 2 RTD-PAC (2726308) IB IL TEMP 2 UTH-PAC (2861386)	none
IL_TEMP_4_8_RTD	Function block for the parameterization and operation of the IB IL TEMP 4/8 RTD and IB IL TEMP 4/8 RTD EF Inline modules.	2.25	IB IL TEMP 4/8 RTD-PAC (2863915) IB IL TEMP 4/8 RTD/EF-PAC (2897402)	none
IL_TEMP_8_UTH_RTD	The function block simplifies the readout of measured values and the writing of the cold junction value. In addition, diagnostic messages, fault codes, terminal firmware, as well as the parameterization of the individual channels are displayed.	1.00	IB IL TEMP 8 UTH/RTD-PAC (2701000)	none
IL_TEMP_HEI_Server	Function block for reading the communication structure.	1.00	IB IL TEMP 4 UTH HEI 1 DO4 (terminated) IB IL TEMP 6 UTH HEI 1 DO6 (terminated) IB IL TEMP 8 UTH HEI 1 DO8 (terminated)	none
IL_TEMP_HEI_ChConf	Function block for the parameterization and operation of the IB IL TEMP 8 UTH HEI 1 DO8, IB IL TEMP 4 UTH HEI 1 DO4, and IB IL TEMP 6 RTD HEI 1 DO6 Inline modules.	1.00	IB IL TEMP 4 UTH HEI 1 DO4 (terminated) IB IL TEMP 6 UTH HEI 1 DO6 (terminated) IB IL TEMP 8 UTH HEI 1 DO8 (terminated)	none

IL_TEMP_HEI_ChVal	Function block for the channel selection of the IB IL TEMP 8 UTH HEI 1 DO8, IB IL TEMP 4 UTH HEI 1 DO4, and IB IL TEMP 6 RTD HEI 1 DO6 Inline modules.	1.00	IB IL TEMP 4 UTH HEI 1 DO4 (terminated) IB IL TEMP 6 UTH HEI 1 DO6 (terminated) IB IL TEMP 8 UTH HEI 1 DO8 (terminated)	none
IL_TempCon	Function block for implementing communication between the controller and the IB IL TEMPCON RTD/UTH temperature controller terminal.	1.00	IB IL TEMP 8 UTH/RTD-PAC (2701000)	none
IL_TempCon_C	Non-editable function block for cyclically reading the values of the IB IL TEMPCON RTD/UTH temperature controller terminal.	1.00	IB IL TEMP 8 UTH/RTD-PAC (2701000)	none
IL_TempCon_Conf	Editable function block for cyclically reading the values of the IB IL TEMPCON RTD/UTH temperature controller terminal.	1.00	IB IL TEMP 8 UTH/RTD-PAC (2701000)	none
IL_TempCon_P	Non-editable function block for parameterization of the IB IL TEMPCON RTD/UTH temperature controller terminal.	1.00	IB IL TEMP 8 UTH/RTD-PAC (2701000)	none
IL_TempCon_Para	Editable function block for parameterization of the IB IL TEMPCON RTD/UTH temperature controller terminal.	1.00	IB IL TEMP 8 UTH/RTD-PAC (2701000)	none

3.50 TouchDisplay

Function block	Description	Version	Supported articles	License
TD_1030T	Function block for the visualization and parameterization of the Minitouch TD 1030T.	1.00	TD 1030T (2701257)	none

3.51 UpsIqBasic

Function block	Description	Version	Supported articles	License
UpsIq_Tcp	Function block for reading basic parameters from QUINT UPS and TRIO UPS devices via a connection to a controller's Ethernet interface and a COMSERVER.	1.04	QUINT-UPS/24DC/24DC/5 (2320212) QUINT-UPS/24DC/24DC/10 (2320225) QUINT-UPS/24DC/24DC/20 (2320238) QUINT-UPS/24DC/24DC/40 (2320241) QUINT-UPS/24DC/24DC/5/1.3AH (2320254) QUINT-UPS/24DC/24DC/10/3.4AH (2320267) QUINT-UPS/1AC/1AC/500VA (2320270) TRIO-UPS/1AC/24DC/5 (2866611)	none
UpsIq_Uni	Function block for reading basic parameters from QUINT UPS IQ and TRIO UPS devices via a connection to the IB IL RS-UNI Inline terminal.	1.05	"	none
UpsIq_Onboard	Function block for reading basic parameters from QUINT UPS and TRIO UPS devices via a connection to a controller's RS-232 onboard interface.	1.04	"	none

3.52 WirelessTechnology

Function block	Description	Version	Supported articles	License
FL_BT_EPA	Function block for handling communication between the controller and the FL BT EPA.	1.04	FL BT EPA (2692788)	none
FL_BT_Diag	Function block for additional diagnostics.	1.00	"	none
FL_BT_Security	Function block for setting the security parameters of the FL BT EPA module.	1.00	"	none
FL_BT_COEX_WLAN	Settings to optimize parallel operation of WLAN and Bluetooth can be made using this function block.	1.10	"	none
FL_BT_Connection	Function block for connecting the FL BT EPA module to a wireless device.	1.00	"	none
FL_WLAN_EPA	Function block for handling communication between the controller and the FL WLAN EPA device.	1.04	FL WLAN EPA (2692791)	none
FL_WLAN_Diag	Function block for additional diagnostics.	1.00	"	none
FL_WLAN_Security	Function block for setting the security parameters of the FL WLAN EPA module.	1.00	"	none
FL_WLAN_Setting	Function block for setting the security parameters of the FL WLAN EPA module.	1.00	"	none
FL_WLAN_Connection	Function block for connecting the FL WLAN EPA module to a wireless device.	1.00	"	none
BT_FL_MOD_IO_AP	Function block for the Modbus access point.	1.00	FL BT MOD IO AP (2884758)	none
BT_FLM_DIO_8_8	Function block for the Fieldline Bluetooth device.	1.01	FLM BT DIO 8/8 M12 (2736767)	none
BT_FLM_DI_16	Function block for the Fieldline Bluetooth device.	1.00	FLM BT DI 16 M12 (2693208)	none
BT_ILB_ADIO_2_2	Function block for the Inline Block Bluetooth device.	1.01	ILB BT ADIO 2/2/16/16 (2884282)	none
BT_FL_MOD_IO_DIAG	Function block for additional diagnostics.	1.10	FL BT MOD IO AP (2884758) FLM BT DIO 8/8 M12 (2736767) FLM BT DI 16 M12 (2693208) ILB BT ADIO 2/2/16/16 (2884282)	none