

SILworX Standard Blocks

Manual

SAFETY
NONSTOP



All HIMA products mentioned in this manual are protected by the HIMA trademark. Unless noted otherwise, this also applies to other manufacturers and their respective products referred to herein.

HIMax®, HIMatrix®, SILworX®, XMR® and FlexSILon® are registered trademarks of HIMA Paul Hildebrandt GmbH.

All of the instructions and technical specifications in this manual have been written with great care and effective quality assurance measures have been implemented to ensure their validity. For questions, please contact HIMA directly. HIMA appreciates any suggestion on which information should be included in the manual.

Equipment subject to change without notice. HIMA also reserves the right to modify the written material without prior notice.

For further information, refer to the HIMA DVD and our website at <http://www.hima.de> and <http://www.hima.com>.

© Copyright 2015, HIMA Paul Hildebrandt GmbH

All rights reserved

Contact

HIMA contact details:

HIMA Paul Hildebrandt GmbH

P.O. Box 1261

68777 Brühl, Germany

Phone: +49 6202 709-0

Fax: +49 6202 709-107

E-mail: info@hima.com

Revision	Change	Type of change	
		Technical	Editorial
7.10	New edition with content from the online help	x	x

Table of Contents

1 Introduction	15
1.1 ENO Behavior of Function Blocks and Functions	15
1.2 POU Instance with EN Input and ENO Output	15
1.3 Recommendations for using EN and ENO	16
2 BISTABLE Sub-Library	17
2.1 RS	17
2.1.1 Inputs	17
2.1.2 Outputs	18
2.1.3 Implementation in Structured Text (ST)	18
2.2 SR	19
2.2.1 Inputs	19
2.2.2 Outputs	20
2.2.3 Implementation in Structured Text (ST)	20
2.3 SEMA	21
2.3.1 Inputs	21
2.3.2 Outputs	22
2.3.3 Implementation in Structured Text (ST)	22
2.3.4 Program Flow Chart	23
3 BITSTR Sub-Library	25
3.1 AND	25
3.1.1 Inputs	25
3.1.2 Outputs	26
3.1.3 Implementation in Structured Text (ST)	26
3.2 NOT	27
3.2.1 Inputs	27
3.2.2 Outputs	28
3.2.3 Implementation in Structured Text (ST)	28
3.3 OR	29
3.3.1 Inputs	29
3.3.2 Outputs	30
3.3.3 Implementation in Structured Text (ST)	30

3.4 ROL	31
3.4.1 Inputs	31
3.4.2 Outputs	31
3.5 ROR	32
3.5.1 Inputs	32
3.5.2 Outputs	32
3.6 SHL	33
3.6.1 Inputs	33
3.6.2 Outputs	33
3.7 SHR	34
3.7.1 Inputs	34
3.7.2 Outputs	34
3.8 XOR	35
3.8.1 Inputs	35
3.8.2 Outputs	35
3.8.3 Implementation in Structured Text (ST)	36
3.8.4 Truth table for XOR with two inputs	36
3.8.5 Truth table for XOR with four inputs	36
4 COMPARE Sub-Library	37
4.1 ANY_REAL Behavior	37
4.2 EQ (Equal)	37
4.2.1 Inputs	37
4.2.2 Outputs	38
4.2.3 Implementation in Structured Text (ST)	38
4.3 GE (Greater or Equal)	39
4.3.1 Inputs	39
4.3.2 Outputs	40
4.3.3 Implementation in Structured Text (ST)	40
4.4 GT (Greater Than)	41
4.4.1 Inputs	41
4.4.2 Outputs	42
4.4.3 Implementation in Structured Text (ST)	42

4.5 LE (Less or Equal)	43
4.5.1 Inputs	43
4.5.2 Outputs	44
4.5.3 Implementation in Structured Text (ST)	44
4.6 LT (Less Than)	45
4.6.1 Inputs	45
4.6.2 Outputs	46
4.6.3 Implementation in Structured Text (ST)	46
4.7 NE (Not Equal)	47
4.7.1 Behavior if the input is NaN	47
4.7.2 Inputs	47
4.7.3 Outputs	48
4.7.4 Implementation in Structured Text (ST)	48
5 CONVERT Sub-Library	49
5.1 ANY_REAL Behavior	49
5.2 ATOBOOL (Any to BOOL)	49
5.2.1 Inputs	49
5.2.2 Outputs	50
5.2.3 Conversion Within the Ranges of Values	50
5.2.4 Conversion Outside the Ranges of Values	51
5.3 ATOBYTE (Any to BYTE)	52
5.3.1 Inputs	52
5.3.2 Outputs	53
5.3.3 Conversion Within the Ranges of Values	54
5.3.4 Conversion Outside the Ranges of Values	55
5.3.5 Conversion of LREAL to ANYBIT and ANYINT	55
5.3.5.1 LREAL_TO_ANYBIT	55
5.3.5.2 LREAL_TO_ANYINT	56
5.4 ATOWORD (Any to WORD)	57
5.4.1 Inputs	57
5.4.2 Outputs	58

5.4.3 Conversion Within the Ranges of Values	59
5.4.4 Conversion Outside the Ranges of Values	60
5.5 ATODWORD (Any to Double WORD)	61
5.5.1 Inputs	61
5.5.2 Outputs	62
5.5.3 Conversion Within the Ranges of Values	63
5.5.4 Conversion Outside the Ranges of Values	64
5.6 ATOLWORD (Any to Long WORD)	65
5.6.1 Inputs	65
5.6.2 Outputs	66
5.7 ATOUSINT (Any to Unsigned Single INT)	67
5.7.1 Inputs	67
5.7.2 Outputs	68
5.7.3 Conversion Within the Ranges of Values	69
5.7.4 Conversion Outside the Ranges of Values	70
5.8 ATOSINT (Any to Single INT)	71
5.8.1 Inputs	71
5.8.2 Outputs	72
5.8.3 Conversion Within the Ranges of Values	74
5.8.4 Conversion Outside the Ranges of Values	74
5.9 ATOUINT (Any to Unsigned INT)	75
5.9.1 Inputs	75
5.9.2 Outputs	76
5.9.3 Conversion Within the Ranges of Values	77
5.9.4 Conversion Outside the Ranges of Values	78
5.10 ATOINT (Any to INT)	79
5.10.1 Inputs	79
5.10.2 Outputs	80
5.10.3 Conversion Within the Ranges of Values	81
5.10.4 Conversion Outside the Ranges of Values	82
5.11 ATOUDINT (Any to Unsigned Double INT)	83
5.11.1 Inputs	83

5.11.2 Outputs	84
5.11.3 Conversion Within the Ranges of Values	85
5.11.4 Conversion Outside the Ranges of Values	86
5.12 ATODINT (Any to Double INT)	87
5.12.1 Inputs	87
5.12.2 Outputs	88
5.12.3 Conversion Within the Ranges of Values	89
5.12.4 Conversion Outside the Ranges of Values	90
5.13 ATOULINT (Any to Unsigned Long INT)	91
5.13.1 Inputs	91
5.13.2 Outputs	92
5.14 ATOLINT (Any to Long INT)	94
5.14.1 Inputs	94
5.14.2 Outputs	94
5.15 ATOTIME (Any to TIME)	96
5.15.1 Inputs	96
5.15.2 Outputs	96
5.15.3 Conversion Within the Ranges of Values	97
5.15.4 Conversion Outside the Ranges of Values	98
5.16 ATOREAL (Any to REAL)	99
5.16.1 Inputs	99
5.16.2 Outputs	100
5.16.3 Converting the Ranges of Values	101
5.17 ATOLREAL (Any to Long REAL)	102
5.17.1 Inputs	102
5.17.2 Outputs	103
5.17.3 Converting the Ranges of Values	103
5.18 TRUNC	104
5.18.1 Inputs	104
5.18.2 Outputs	105
5.18.3 Conversion Rules	105

5.19 PACK	106
5.19.1 Inputs	106
5.19.2 Outputs	107
5.20 UNPACK2	108
5.20.1 Inputs	108
5.20.2 Outputs	109
5.21 UNPACK4	110
5.21.1 Inputs	110
5.21.2 Outputs	111
5.22 UNPACK8	112
5.22.1 Inputs	112
5.22.2 Outputs	113
5.23 BIT_PACK	114
5.23.1 Inputs	114
5.23.2 Outputs	115
5.24 BIT_UNPACK	116
5.24.1 Inputs	116
5.24.2 Outputs	117
6 COUNTER Sub-Library	119
6.1 Values Allowed for PV	119
6.2 CTD (Counter Down)	119
6.2.1 Inputs	119
6.2.2 Outputs	120
6.2.3 Implementation in Structured Text (ST)	121
6.3 CTU (Counter Up)	121
6.3.1 Inputs	121
6.3.2 Outputs	122
6.3.3 Implementation in Structured Text (ST)	123
6.4 CTUD (Counter Up/Down)	123
6.4.1 Inputs	123
6.4.2 Outputs	125
6.4.3 Implementation in Structured Text (ST)	125

7 EDGE Sub-Library	127
7.1 F_TRIG	127
7.1.1 Inputs	127
7.1.2 Outputs	127
7.1.3 Implementation in Structured Text (ST)	127
7.2 R_TRIG	128
7.2.1 Inputs	128
7.2.2 Outputs	129
7.2.3 Implementation in Structured Text (ST)	129
8 NUMERIC Sub-Library	131
8.1 ABS (Absolute Value)	131
8.1.1 Inputs	131
8.1.2 Outputs	132
8.2 SQRT (Square Root)	133
8.2.1 Inputs	133
8.2.2 Outputs	134
8.3 LN (Natural Logarithm)	135
8.3.1 Inputs	135
8.3.2 Outputs	136
8.4 LOG (Base-10 Logarithm)	137
8.4.1 Inputs	137
8.4.2 Outputs	138
8.5 EXP (Exponential Function ex)	139
8.5.1 Inputs	139
8.5.2 Outputs	140
8.6 SIN (Trigonometric Function: Sine)	141
8.6.1 Inputs	141
8.6.2 Outputs	142
8.7 COS (Trigonometric Function: Cosine)	143
8.7.1 Inputs	143
8.7.2 Outputs	144

8.8 TAN (Trigonometric Function: Tangent)	145
8.8.1 Inputs	145
8.8.2 Outputs	146
8.9 ASIN (Trigonometric Function: Arcus Sine)	147
8.9.1 Inputs	147
8.9.2 Outputs	148
8.10 ACOS (Trigonometric Function: Arc Cosine)	149
8.10.1 Inputs	149
8.10.2 Outputs	150
8.11 ATAN (Trigonometric Function: Arc Tangent)	151
8.11.1 Inputs	151
8.11.2 Outputs	152
8.12 ADD (Addition)	153
8.12.1 Inputs	153
8.12.2 Outputs	154
8.12.3 Implementation in Structured Text (ST)	154
8.13 MUL (Multiplication)	155
8.13.1 Inputs	155
8.13.2 Outputs	156
8.13.3 Implementation in Structured Text (ST)	156
8.14 SUB (Subtraction)	157
8.14.1 Inputs	157
8.14.2 Outputs	158
8.14.3 Implementation in Structured Text (ST)	158
8.15 DIV (Division)	159
8.15.1 Inputs	159
8.15.2 Outputs	159
8.15.3 Behavior with REAL and LREAL	160
8.15.4 Implementation in Structured Text (ST)	160
8.16 MOD (Modulo)	161
8.16.1 Inputs	161
8.16.2 Outputs	162

8.16.3 Implementation in Structured Text (ST)	162
8.17 EXPT (Exponentiate)	163
8.17.1 Inputs	163
8.17.2 Outputs	164
8.17.3 Implementation in Structured Text (ST)	164
8.18 MOVE	165
8.18.1 Inputs	165
8.18.2 Outputs	165
8.18.3 Application Example	166
8.19 CHS (Change Sign)	167
8.19.1 Inputs	167
8.19.2 Outputs	168
8.19.3 Special Cases	168
8.19.3.1 IN as an Unsigned Integer Data Type	168
8.19.3.2 IN as the Largest Negative Signed Integer:	168
8.20 CHK_REAL	169
8.20.1 Inputs	169
8.20.2 Outputs	170
9 SELECT Sub-Library	171
9.1 SEL (Select)	171
9.1.1 Inputs	171
9.1.2 Outputs	172
9.2 MAX (Maximum Value)	173
9.2.1 Inputs	173
9.2.2 Outputs	174
9.3 MIN (Minimum Value)	175
9.3.1 Inputs	175
9.3.2 Outputs	176
9.4 LIMIT (Limiter)	177
9.4.1 Inputs	177
9.4.2 Outputs	178

9.4.3 Exceptions for REAL/LREAL Type Conversion	178
9.4.4 Implementation in Structured Text (ST)	178
9.5 MUX (Multiplexer)	179
9.5.1 Inputs	179
9.5.2 Outputs	180
9.5.3 Implementation in Structured Text (ST)	180
10 TIME Sub-Library	181
10.1 DIV_TIME	181
10.1.1 Inputs	181
10.1.2 Outputs	182
10.2 MUL_TIME	183
10.2.1 Inputs	183
10.2.2 Outputs	184
10.3 ADD_TIME	185
10.3.1 Inputs	185
10.3.2 Outputs	186
10.4 SUB_TIME	187
10.4.1 Inputs	187
10.4.2 Outputs	187
11 TIMER Sub-Library	189
11.1 General Timer Behavior	189
11.2 Behavior of Timers with Retain Feature	189
11.3 TOF	190
11.3.1 Inputs	190
11.3.2 Outputs	191
11.3.3 Graphic on TOF Behavior	192
11.4 TON	193
11.4.1 Inputs	193
11.4.2 Outputs	194
11.4.3 Graphic on TON Behavior	195
11.5 TP	196
11.5.1 Inputs	196

11.5.2 Outputs	197
11.5.3 Graphic on TP Behavior	198
12 Sequential Function Charts (SFC)	199
12.1 Working with Step Sequences	199
12.2 Rules for SFC Connections	200
12.3 Initial Step	200
12.4 Step	201
12.4.1 Definition According to IEC 61131-3	201
12.5 Transition	202
12.5.1 Definition According to IEC 61131-3	202
12.6 Action	203
12.6.1 SFC Action Fields	203
12.6.2 Behavior of SFC Actions	203
12.6.3 Functions and SFC Elements	203
12.6.4 Differences Between ELOP II and SILworX	204
12.6.5 SFC Action Name (AN)	204
12.7 Creating SFC Connectors	205
12.8 Action Qualifier (AQ)	206
12.8.1 Function of the Action Qualifier	208
12.8.2 Time Period (T)	208
12.8.3 Action Qualifier D+R	209
12.8.4 Action Qualifier DS+R	209
12.8.5 Action Qualifier L+R	210
12.8.6 Action Qualifier N+R	210
12.8.7 Action Qualifier P+R	211
12.8.8 Action Qualifier P0+R	211
12.8.9 Action Qualifier P1+R	212
12.8.10 Action Qualifier PN+R	212
12.8.11 Action Qualifier S+R	213
12.8.12 Action Qualifier SD+R	213
12.8.13 Action Qualifier SL+R	214

1 Introduction

The IEC 61131 standard for programmable systems describes the behavior and characteristics of standard functions and standard function blocks (StdLib) available to the user.

This document describes the standard library in accordance with IEC 61131-3:2003 and its implementation in SILworX. The following information is provided for each function and each function block:

- Details about the function and behavior.
- Explanation of the inputs (name, type, range of values if restricted).
- Explanation of the outputs (name, type, range of values if restricted).
- Details about the extendibility
- Explanation of the ENO behavior.
- Characteristics that, according to the standard, depend on the implementation, or that imply an extention of or a deviation from the standard.

For all the functions and function blocks of the standard library, the information and details mentioned above are also available in SILworX as Online Help.

1.1 ENO Behavior of Function Blocks and Functions

All standard function blocks and functions (POUs) also have the [EN](#) input and the [ENO](#) output, that may be displayed.

As long as the EN input is not connected, it is set to TRUE. The ENO output corresponds to the value of EN and is TRUE, if the POU was properly processed.

Additionally, ENO can be set to FALSE through the internal POU error detection if an error occurred during POU processing. Not all POUs, however, supports this ENO behavior.

1.2 POU Instance with EN Input and ENO Output

To display the EN input and ENO output associated with each POU instance in the Drawing Area, right-click the FBD Editor and select **Show EN/ENO** from the context menu. Since EN and ENO are located above the usual inputs and outputs, they cannot be displayed there is not enough room above the POU instance. Both interfaces have the sequence number 0.

EN and ENO can only be hidden if both interfaces are not connected to the logic.

A POU is only processed, if EN = TRUE. If a POU with EN = FALSE is deactivated, ENO is also set to FALSE.

1.3 Recommendations for using EN and ENO

EN	FBS¹	Fkt²	Description	
TRUE	X	X	<ul style="list-style-type: none"> The POU is executed. <p>If the POU supports error detection:</p> <ul style="list-style-type: none"> ENO = TRUE: The logic was properly processed. ENO = FALSE: An error occurred while executing the POU. <p>If the POU does not support error detection:</p> <ul style="list-style-type: none"> ENO = TRUE: The ENO output is always TRUE. 	
FALSE	X	-	<ul style="list-style-type: none"> The POU is not executed. No new output values are calculated. All outputs except for ENO retain the previous values. ENO = FALSE 	<p>If a POU is not executed because EN = FALSE, the logic connected to the OUTn output behaves as follow:</p> <ul style="list-style-type: none"> Variables directly connected to OUTn are not rewritten. They retain the value that they had when EN = TRUE. This also applies if multiple variables are consecutively connected in a chain. Function blocks or functions directly connected to OUTn actively read out the value of OUTn. OUTn is the output of a function block: The connected logic continues to run with the last value used for OUTn. OUTn is the output of a function: The connected logic continues to run with the default value of OUTn.
	-	X	<ul style="list-style-type: none"> The POU is not executed. No new output values are calculated. Except for ENO, all other outputs are reset to the default values of the corresponding data type. ENO = FALSE 	

(1) Function block, (2) Function

2 BISTABLE Sub-Library

2.1 RS

The RS flip flop is a bistable function block with overriding reset.

2.1.1 Inputs

Input	Data type	Description	
S	BOOL	A rising edge at S (Set) sets the Q1 output to TRUE. A falling edge at S has no effect on the Q1 output.	
R1	BOOL	The R1 input is dominant: If R1 is permanently set to TRUE, S has no effect on the Q1 output.	
		R1 = FALSE S = FALSE → TRUE	A rising edge at S sets the Q1 output to TRUE.
		R1 = TRUE S = FALSE → TRUE	A rising edge at S has no effect on the Q1 output.
		R1 = TRUE → FALSE S = TRUE	If R1 = FALSE, then Q = TRUE.
EN	BOOL	R1 = FALSE → TRUE S = TRUE	A rising edge at R1 resets the output: Q1 = FALSE.
		EN is hidden by default. If EN is connected, it is always TRUE. Show EN/ENO: Right-click a function block or function, and select Show EN/ENO from the context menu. As soon as EN or ENO is connected, it can no longer be hidden.	
		TRUE	The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle. ENO Output: The output is TRUE and can be set to FALSE separately based on the internal error detection.
		FALSE	The function block or function is not processed by the user program. The following variables and value field assignment are not assigned. The outputs of a function block display the results of the user program cycle that was lastly processed. The outputs of a function display the default values defined for the data type. As long as the function block or function has not been completely processed, the outputs retain the default values defined for the data type. ENO Output: The output is also set to FALSE.

2.1.2 Outputs

Output	Data type	Description	
Q1	BOOL	TRUE	After a signal change FALSE → TRUE at S.
		FALSE	<ul style="list-style-type: none"> • Default setting after a program start or • after a reset ((→ R1)).
<u>ENO</u>	BOOL	TRUE	If EN is TRUE and the current cycle of the function block or function was processed without errors, ENO is also TRUE. The function cannot result in an error.
		FALSE	Only if EN is also FALSE or while the function block or function is being processed.

2.1.3 Implementation in Structured Text (ST)

Q1 := NOT R1 AND (S OR Q1)

2.2 SR

The SR flip flop is a bistable function block with overriding set.

2.2.1 Inputs

Input	Data type	Description			
S1	BOOL	<p>A rising edge at S1 (Set) sets the Q1 output to TRUE. A falling edge at S1 has no effect on the Q1 output.</p> <p>The S1 input is dominant:</p> <p>If S1 is permanently set to TRUE, R has no effect on the Q1 output.</p>			
R	BOOL	S1 = FALSE R = FALSE → TRUE	A rising edge at R (Reset) resets the output: Q1 = FALSE.		
		S1 = TRUE R = FALSE → TRUE	A rising edge at R has no effect on the Q1 output.		
		S1 = TRUE → FALSE R = TRUE	If S1 = FALSE, then Q1 = FALSE.		
		S1 = FALSE → TRUE R = TRUE	A rising edge at S1 sets the Q1 output to TRUE.		
<u>EN</u>	BOOL	<p>EN is hidden by default. If EN is connected, it is always TRUE.</p> <p>Show EN/ENO: Right-click a function block or function, and select Show EN/ENO from the context menu.</p> <p>As soon as EN or ENO is connected, it can no longer be hidden.</p>			
		<table border="1"> <tr> <td>TRUE</td> <td> <p>The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle.</p> <p>ENO Output: The output is TRUE and can be set to FALSE separately based on the internal error detection.</p> </td> </tr> <tr> <td>FALSE</td> <td> <p>The function block or function is not processed by the user program. The following variables and value field assignment are not assigned.</p> <p>The outputs of a function block display the results of the user program cycle that was lastly processed.</p> <p>The outputs of a function display the default values defined for the data type.</p> <p>As long as the function block or function has not been completely processed, the outputs retain the default values defined for the data type.</p> <p>ENO Output: The output is also set to FALSE.</p> </td> </tr> </table>	TRUE	<p>The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle.</p> <p>ENO Output: The output is TRUE and can be set to FALSE separately based on the internal error detection.</p>	FALSE
TRUE	<p>The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle.</p> <p>ENO Output: The output is TRUE and can be set to FALSE separately based on the internal error detection.</p>				
FALSE	<p>The function block or function is not processed by the user program. The following variables and value field assignment are not assigned.</p> <p>The outputs of a function block display the results of the user program cycle that was lastly processed.</p> <p>The outputs of a function display the default values defined for the data type.</p> <p>As long as the function block or function has not been completely processed, the outputs retain the default values defined for the data type.</p> <p>ENO Output: The output is also set to FALSE.</p>				

2.2.2 Outputs

Output	Data type	Description	
Q1	BOOL	TRUE	After a signal change FALSE → TRUE at S1.
		FALSE	<ul style="list-style-type: none"> • Default setting after a program start or • after a reset ($\rightarrow R$)
<u>ENO</u>	BOOL	TRUE	If EN is TRUE and the current cycle of the function block or function was processed without errors, ENO is also TRUE. The function cannot result in an error.
		FALSE	Only if EN is also FALSE or while the function block or function is being processed.

2.2.3 Implementation in Structured Text (ST)

Q1 := S1 OR (NOT R AND Q1);

2.3 SEMA

This function block is used to delay the output of a cycle.

2.3.1 Inputs

Input	Data type	Description	
CLAIM	BOOL	TRUE	<ul style="list-style-type: none"> After the rising edge, the internal variable X is set to TRUE during the first program cycle. (*) In the following program cycle, BUSY = X = TRUE. (*) <p>(*) This is also the case if RELEASE = TRUE.</p>
		TRUE → FALSE	A falling edge at the CLAIM input has no effect on the BUSY output.
		FALSE	CLAIM has no effect on the internal variable X and the BUSY output.
RELEASE	BOOL	TRUE	<ul style="list-style-type: none"> If CLAIM = FALSE: X = FALSE BUSY = FALSE If CLAIM = TRUE: RELEASE has no effect on the internal variable X and the BUSY output. If CLAIM = TRUE → FALSE: As soon as CLAIM is set to FALSE, BUSY is set to FALSE.
		FALSE	RELEASE has no effect on the internal variable X and the BUSY output.

Input	Data type	Description			
EN	BOOL	EN is hidden by default. If EN is connected, it is always TRUE. Show EN/ENO: Right-click a function block or function, and select Show EN/ENO from the context menu. As soon as EN or ENO is connected, it can no longer be hidden.			
		<table border="1"> <tr> <td>TRUE</td><td>The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle. ENO Output: The output is TRUE and can be set to FALSE separately based on the internal error detection.</td></tr> <tr> <td>FALSE</td><td>The function block or function is not processed by the user program. The following variables and value field assignment are not assigned. The outputs of a function block display the results of the user program cycle that was lastly processed. The outputs of a function display the default values defined for the data type. As long as the function block or function has not been completely processed, the outputs retain the default values defined for the data type. ENO Output: The output is also set to FALSE.</td></tr> </table>	TRUE	The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle. ENO Output: The output is TRUE and can be set to FALSE separately based on the internal error detection.	FALSE
TRUE	The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle. ENO Output: The output is TRUE and can be set to FALSE separately based on the internal error detection.				
FALSE	The function block or function is not processed by the user program. The following variables and value field assignment are not assigned. The outputs of a function block display the results of the user program cycle that was lastly processed. The outputs of a function display the default values defined for the data type. As long as the function block or function has not been completely processed, the outputs retain the default values defined for the data type. ENO Output: The output is also set to FALSE.				

2.3.2 Outputs

Output	Data type	Description	
BUSY	BOOL	See the inputs <i>CLAIM</i> and <i>RELEASE</i> .	
ENO	BOOL	TRUE	If EN is TRUE and the current cycle of the function block or function was processed without errors, ENO is also TRUE. The function cannot result in an error.
		FALSE	Only if EN is also FALSE or while the function block or function is being processed.

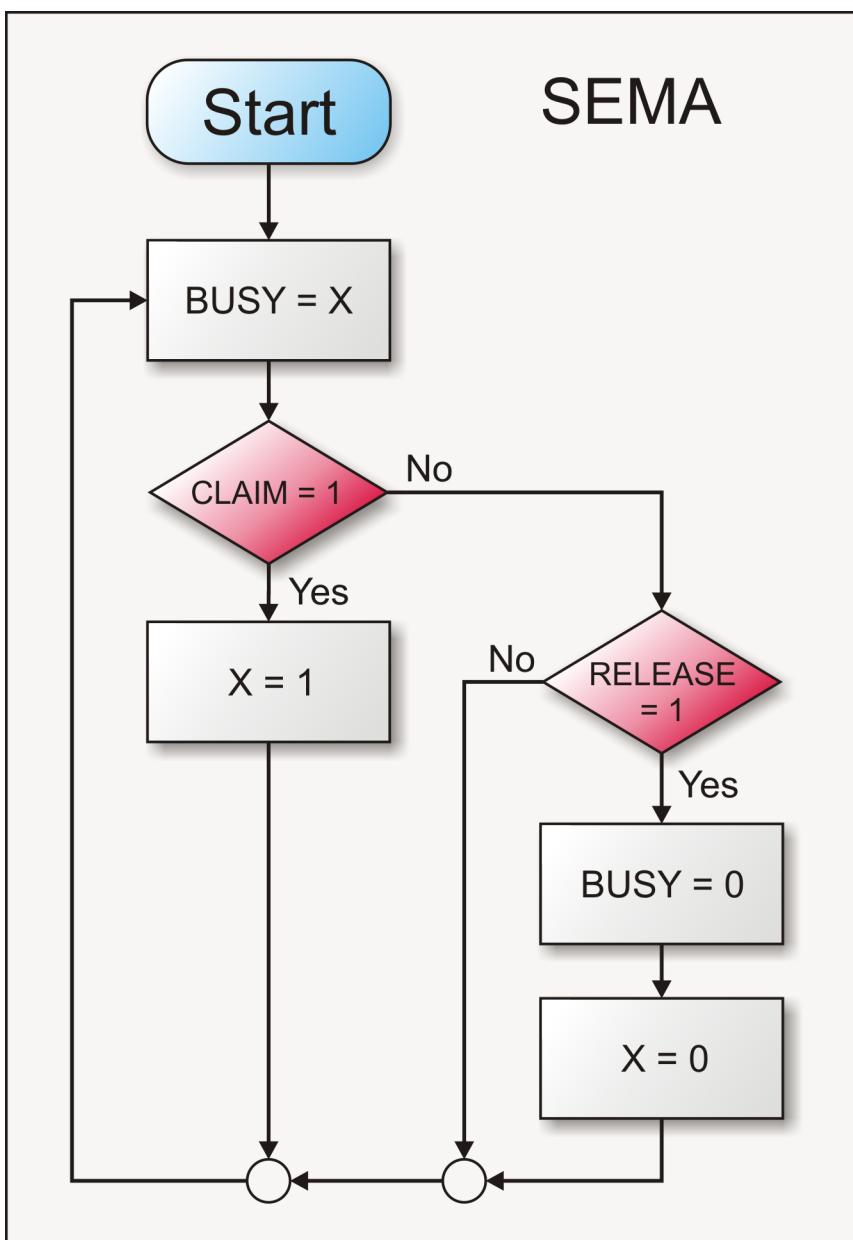
2.3.3 Implementation in Structured Text (ST)

```

VAR X : BOOL := 0; END_VAR
BUSY := X;
IF CLAIM THEN X := 1;
ELSIF RELEASE THEN
  BUSY := 0; X := 0;
END_IF

```

2.3.4 Program Flow Chart



3 BITSTR Sub-Library

3.1 AND

The function performs a logical AND operation on the BOOL data type, and, with the data types BYTE through LWORD, it performs a bitwise AND operation on the inputs.

3.1.1 Inputs

Input	Data type	Description				
IN1	ANY_BIT	Input IN1				
IN2	ANY_BIT	Input IN2				
Extendible	ANY_BIT	Up to IN16				
EN	BOOL	<p>EN is hidden by default. If EN is connected, it is always TRUE. Show EN/ENO: Right-click a function block or function, and select Show EN/ENO from the context menu.</p> <p>As soon as EN or ENO is connected, it can no longer be hidden.</p> <table border="1"> <tr> <td>TRUE</td><td> <p>The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle.</p> <p>ENO Output: The output is TRUE and can be set to FALSE separately based on the internal error detection.</p> </td></tr> <tr> <td>FALSE</td><td> <p>The function block or function is not processed by the user program. The following variables and value field assignment are not assigned.</p> <p>The outputs of a function block display the results of the user program cycle that was lastly processed.</p> <p>The outputs of a function display the default values defined for the data type.</p> <p>As long as the function block or function has not been completely processed, the outputs retain the default values defined for the data type.</p> <p>ENO Output: The output is also set to FALSE.</p> </td></tr> </table>	TRUE	<p>The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle.</p> <p>ENO Output: The output is TRUE and can be set to FALSE separately based on the internal error detection.</p>	FALSE	<p>The function block or function is not processed by the user program. The following variables and value field assignment are not assigned.</p> <p>The outputs of a function block display the results of the user program cycle that was lastly processed.</p> <p>The outputs of a function display the default values defined for the data type.</p> <p>As long as the function block or function has not been completely processed, the outputs retain the default values defined for the data type.</p> <p>ENO Output: The output is also set to FALSE.</p>
TRUE	<p>The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle.</p> <p>ENO Output: The output is TRUE and can be set to FALSE separately based on the internal error detection.</p>					
FALSE	<p>The function block or function is not processed by the user program. The following variables and value field assignment are not assigned.</p> <p>The outputs of a function block display the results of the user program cycle that was lastly processed.</p> <p>The outputs of a function display the default values defined for the data type.</p> <p>As long as the function block or function has not been completely processed, the outputs retain the default values defined for the data type.</p> <p>ENO Output: The output is also set to FALSE.</p>					

3.1.2 Outputs

Output	Data type	Description	
OUT	ANY_BIT	Performs a logical AND operation on the BOOL data type, and a bitwise AND operation on the data types BYTE to LWORD at the inputs IN1 through INn.	
<u>ENO</u>	BOOL	TRUE	If EN is TRUE and the current cycle of the function block or function was processed without errors, ENO is also TRUE. The function cannot result in an error.
		FALSE	Only if EN is also FALSE or while the function block or function is being processed.

3.1.3 Implementation in Structured Text (ST)

```
OUT := IN1 AND IN2 AND .. AND INn;
```

3.2 NOT

This function performs a logical negation on the BOOL data type, and forms the bitwise complement of the input for the data types BYTE to LWORD.

3.2.1 Inputs

Input	Data type	Description				
IN1	ANY_BIT	Input IN1				
EN	BOOL	<p>EN is hidden by default. If EN is connected, it is always TRUE.</p> <p>Show EN/ENO: Right-click a function block or function, and select Show EN/ENO from the context menu.</p> <p>As soon as EN or ENO is connected, it can no longer be hidden.</p> <table border="1"> <tr> <td>TRUE</td><td>The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle. ENO Output: The output is TRUE and can be set to FALSE separately based on the internal error detection.</td></tr> <tr> <td>FALSE</td><td>The function block or function is not processed by the user program. The following variables and value field assignment are not assigned. The outputs of a function block display the results of the user program cycle that was lastly processed. The outputs of a function display the default values defined for the data type. As long as the function block or function has not been completely processed, the outputs retain the default values defined for the data type. ENO Output: The output is also set to FALSE.</td></tr> </table>	TRUE	The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle. ENO Output: The output is TRUE and can be set to FALSE separately based on the internal error detection.	FALSE	The function block or function is not processed by the user program. The following variables and value field assignment are not assigned. The outputs of a function block display the results of the user program cycle that was lastly processed. The outputs of a function display the default values defined for the data type. As long as the function block or function has not been completely processed, the outputs retain the default values defined for the data type. ENO Output: The output is also set to FALSE.
TRUE	The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle. ENO Output: The output is TRUE and can be set to FALSE separately based on the internal error detection.					
FALSE	The function block or function is not processed by the user program. The following variables and value field assignment are not assigned. The outputs of a function block display the results of the user program cycle that was lastly processed. The outputs of a function display the default values defined for the data type. As long as the function block or function has not been completely processed, the outputs retain the default values defined for the data type. ENO Output: The output is also set to FALSE.					

3.2.2 Outputs

Output	Data type	Description				
OUT	ANY_BIT	<p>OUT = NOT IN1</p> <p>For the BOOL data type: A logical negation of the IN1 value is performed.</p> <p>For the BYTE, WORD, DWORD, LWORD data types: A bitwise inversion of the IN1 value is performed.</p> <p> To negate the value of an input or output, right-click the interface and select Invert from the context menu. The advantage of using the separate NOT function block is that the processing sequence can be modified by changing the position in the workspace.</p>				
ENO	BOOL	<table border="1"> <tr> <td>TRUE</td><td>If EN is TRUE and the current cycle of the function block or function was processed without errors, ENO is also TRUE. The function cannot result in an error.</td></tr> <tr> <td>FALSE</td><td>Only if EN is also FALSE or while the function block or function is being processed.</td></tr> </table>	TRUE	If EN is TRUE and the current cycle of the function block or function was processed without errors, ENO is also TRUE. The function cannot result in an error.	FALSE	Only if EN is also FALSE or while the function block or function is being processed.
TRUE	If EN is TRUE and the current cycle of the function block or function was processed without errors, ENO is also TRUE. The function cannot result in an error.					
FALSE	Only if EN is also FALSE or while the function block or function is being processed.					

3.2.3 Implementation in Structured Text (ST)

```
OUT := NOT IN;
```

3.3 OR

This function performs a logical OR operation on the BOOL data type, and, with the data types BYTE through LWORD, it performs a bitwise OR operation on the inputs.

3.3.1 Inputs

Input	Data type	Description				
IN1	ANY_BIT	Input IN1				
IN2	ANY_BIT	Input IN2				
Extendible	ANY_BIT	Up to IN16				
EN	BOOL	<p>EN is hidden by default. If EN is connected, it is always TRUE.</p> <p>Show EN/ENO: Right-click a function block or function, and select Show EN/ENO from the context menu.</p> <p>As soon as EN or ENO is connected, it can no longer be hidden.</p> <table border="1"> <tr> <td>TRUE</td><td> <p>The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle.</p> <p>ENO Output: The output is TRUE and can be set to FALSE separately based on the internal error detection.</p> </td></tr> <tr> <td>FALSE</td><td> <p>The function block or function is not processed by the user program. The following variables and value field assignment are not assigned.</p> <p>The outputs of a function block display the results of the user program cycle that was lastly processed.</p> <p>The outputs of a function display the default values defined for the data type.</p> <p>As long as the function block or function has not been completely processed, the outputs retain the default values defined for the data type.</p> <p>ENO Output: The output is also set to FALSE.</p> </td></tr> </table>	TRUE	<p>The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle.</p> <p>ENO Output: The output is TRUE and can be set to FALSE separately based on the internal error detection.</p>	FALSE	<p>The function block or function is not processed by the user program. The following variables and value field assignment are not assigned.</p> <p>The outputs of a function block display the results of the user program cycle that was lastly processed.</p> <p>The outputs of a function display the default values defined for the data type.</p> <p>As long as the function block or function has not been completely processed, the outputs retain the default values defined for the data type.</p> <p>ENO Output: The output is also set to FALSE.</p>
TRUE	<p>The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle.</p> <p>ENO Output: The output is TRUE and can be set to FALSE separately based on the internal error detection.</p>					
FALSE	<p>The function block or function is not processed by the user program. The following variables and value field assignment are not assigned.</p> <p>The outputs of a function block display the results of the user program cycle that was lastly processed.</p> <p>The outputs of a function display the default values defined for the data type.</p> <p>As long as the function block or function has not been completely processed, the outputs retain the default values defined for the data type.</p> <p>ENO Output: The output is also set to FALSE.</p>					

3.3.2 Outputs

Output	Data type	Description	
OUT	ANY_BIT	<p>OUT = IN1 OR IN2 OR ... INn</p> <p>For the BOOL data type: A logical disjunction (OR gate) of the inputs used is performed.</p> <p>For the BYTE, WORD, DWORD, LWORD data types: A bitwise disjunction (OR gate) of the inputs used is performed.</p>	
<u>ENO</u>	BOOL	TRUE	If EN is TRUE and the current cycle of the function block or function was processed without errors, ENO is also TRUE. The function cannot result in an error.
		FALSE	Only if EN is also FALSE or while the function block or function is being processed.

3.3.3 Implementation in Structured Text (ST)

```
OUT := IN1 OR IN2 OR .. OR INn;
```

3.4 ROL

This function rotates the bits of the value at IN n times to the left. If IN is signed and negative, the rotation direction is inverted.

3.4.1 Inputs

Input	Data type	Description				
IN	ANY_BIT	Input IN				
N	ANY_INT	Number of roll left steps				
EN	BOOL	<p>EN is hidden by default. If EN is connected, it is always TRUE.</p> <p>Show EN/ENO: Right-click a function block or function, and select Show EN/ENO from the context menu.</p> <p>As soon as EN or ENO is connected, it can no longer be hidden.</p> <table border="1"> <tr> <td>TRUE</td><td> The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle. ENO Output: The output is TRUE and can be set to FALSE separately based on the internal error detection. </td></tr> <tr> <td>FALSE</td><td> The function block or function is not processed by the user program. The following variables and value field assignment are not assigned. The outputs of a function block display the results of the user program cycle that was lastly processed. The outputs of a function display the default values defined for the data type. As long as the function block or function has not been completely processed, the outputs retain the default values defined for the data type. ENO Output: The output is also set to FALSE. </td></tr> </table>	TRUE	The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle. ENO Output: The output is TRUE and can be set to FALSE separately based on the internal error detection.	FALSE	The function block or function is not processed by the user program. The following variables and value field assignment are not assigned. The outputs of a function block display the results of the user program cycle that was lastly processed. The outputs of a function display the default values defined for the data type. As long as the function block or function has not been completely processed, the outputs retain the default values defined for the data type. ENO Output: The output is also set to FALSE.
TRUE	The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle. ENO Output: The output is TRUE and can be set to FALSE separately based on the internal error detection.					
FALSE	The function block or function is not processed by the user program. The following variables and value field assignment are not assigned. The outputs of a function block display the results of the user program cycle that was lastly processed. The outputs of a function display the default values defined for the data type. As long as the function block or function has not been completely processed, the outputs retain the default values defined for the data type. ENO Output: The output is also set to FALSE.					

3.4.2 Outputs

Output	Data type	Description				
OUT	ANY_BIT	The IN value is n times shifted bit by bit to the left. The most significant bit, which is shifted out to the left, is rotated to the least significant bit position.				
ENO	BOOL	<table border="1"> <tr> <td>TRUE</td><td>If EN is TRUE and the current cycle of the function block or function was processed without errors, ENO is also TRUE. The function cannot result in an error.</td></tr> <tr> <td>FALSE</td><td>If also EN = FALSE, or if the N input is signed and negative.</td></tr> </table>	TRUE	If EN is TRUE and the current cycle of the function block or function was processed without errors, ENO is also TRUE. The function cannot result in an error.	FALSE	If also EN = FALSE, or if the N input is signed and negative.
TRUE	If EN is TRUE and the current cycle of the function block or function was processed without errors, ENO is also TRUE. The function cannot result in an error.					
FALSE	If also EN = FALSE, or if the N input is signed and negative.					

3.5 ROR

This function rotates the bits of the value at IN n times to the right. If IN is signed and negative, the rotation direction is inverted.

3.5.1 Inputs

Input	Data type	Description				
IN	ANY_BIT	Input IN				
N	ANY_INT	Number of roll right steps				
EN	BOOL	<p>EN is hidden by default. If EN is connected, it is always TRUE. Show EN/ENO: Right-click a function block or function, and select Show EN/ENO from the context menu.</p> <p>As soon as EN or ENO is connected, it can no longer be hidden.</p> <table border="1"> <tr> <td>TRUE</td><td> <p>The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle.</p> <p>ENO Output: The output is TRUE and can be set to FALSE separately based on the internal error detection.</p> </td></tr> <tr> <td>FALSE</td><td> <p>The function block or function is not processed by the user program. The following variables and value field assignment are not assigned.</p> <p>The outputs of a function block display the results of the user program cycle that was lastly processed.</p> <p>The outputs of a function display the default values defined for the data type.</p> <p>As long as the function block or function has not been completely processed, the outputs retain the default values defined for the data type.</p> <p>ENO Output: The output is also set to FALSE.</p> </td></tr> </table>	TRUE	<p>The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle.</p> <p>ENO Output: The output is TRUE and can be set to FALSE separately based on the internal error detection.</p>	FALSE	<p>The function block or function is not processed by the user program. The following variables and value field assignment are not assigned.</p> <p>The outputs of a function block display the results of the user program cycle that was lastly processed.</p> <p>The outputs of a function display the default values defined for the data type.</p> <p>As long as the function block or function has not been completely processed, the outputs retain the default values defined for the data type.</p> <p>ENO Output: The output is also set to FALSE.</p>
TRUE	<p>The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle.</p> <p>ENO Output: The output is TRUE and can be set to FALSE separately based on the internal error detection.</p>					
FALSE	<p>The function block or function is not processed by the user program. The following variables and value field assignment are not assigned.</p> <p>The outputs of a function block display the results of the user program cycle that was lastly processed.</p> <p>The outputs of a function display the default values defined for the data type.</p> <p>As long as the function block or function has not been completely processed, the outputs retain the default values defined for the data type.</p> <p>ENO Output: The output is also set to FALSE.</p>					

3.5.2 Outputs

Output	Data type	Description				
OUT	ANY_BIT	<p>The IN value is n times bitwise shifted to the right.</p> <p>The least significant bit, which is shifted out to the right, is rotated to the most significant bit position.</p>				
ENO	BOOL	<table border="1"> <tr> <td>TRUE</td><td>If EN is TRUE and the current cycle of the function block or function was processed without errors, ENO is also TRUE. The function cannot result in an error.</td></tr> <tr> <td>FALSE</td><td>If also EN = FALSE, or if the N input is signed and negative.</td></tr> </table>	TRUE	If EN is TRUE and the current cycle of the function block or function was processed without errors, ENO is also TRUE. The function cannot result in an error.	FALSE	If also EN = FALSE, or if the N input is signed and negative.
TRUE	If EN is TRUE and the current cycle of the function block or function was processed without errors, ENO is also TRUE. The function cannot result in an error.					
FALSE	If also EN = FALSE, or if the N input is signed and negative.					

3.6 SHL

This function shifts the bits of the value at IN n times to the left and fills in the bits to the right with zeros. If IN is signed and negative, the shifting direction is inverted.

3.6.1 Inputs

Input	Data type	Description				
IN	ANY_BIT	Input IN				
N	ANY_INT	Number of shifting left steps				
EN	BOOL	<p>EN is hidden by default. If EN is connected, it is always TRUE. Show EN/ENO: Right-click a function block or function, and select Show EN/ENO from the context menu.</p> <p>As soon as EN or ENO is connected, it can no longer be hidden.</p> <table border="1"> <tr> <td>TRUE</td> <td>The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle. ENO Output: The output is TRUE and can be set to FALSE separately based on the internal error detection.</td> </tr> <tr> <td>FALSE</td> <td>The function block or function is not processed by the user program. The following variables and value field assignment are not assigned. The outputs of a function block display the results of the user program cycle that was lastly processed. The outputs of a function display the default values defined for the data type. As long as the function block or function has not been completely processed, the outputs retain the default values defined for the data type. ENO Output: The output is also set to FALSE.</td> </tr> </table>	TRUE	The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle. ENO Output: The output is TRUE and can be set to FALSE separately based on the internal error detection.	FALSE	The function block or function is not processed by the user program. The following variables and value field assignment are not assigned. The outputs of a function block display the results of the user program cycle that was lastly processed. The outputs of a function display the default values defined for the data type. As long as the function block or function has not been completely processed, the outputs retain the default values defined for the data type. ENO Output: The output is also set to FALSE.
TRUE	The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle. ENO Output: The output is TRUE and can be set to FALSE separately based on the internal error detection.					
FALSE	The function block or function is not processed by the user program. The following variables and value field assignment are not assigned. The outputs of a function block display the results of the user program cycle that was lastly processed. The outputs of a function display the default values defined for the data type. As long as the function block or function has not been completely processed, the outputs retain the default values defined for the data type. ENO Output: The output is also set to FALSE.					

3.6.2 Outputs

Output	Data type	Description				
OUT	ANY_BIT	The IN value is n times bitwise shifted to the left. The least significant bit is set to 0 after each shifting left step.				
ENO	BOOL	<table border="1"> <tr> <td>TRUE</td> <td>If EN is TRUE and the current cycle of the function block or function was processed without errors, ENO is also TRUE. The function cannot result in an error.</td> </tr> <tr> <td>FALSE</td> <td>If also EN = FALSE, or if the N input is signed and negative.</td> </tr> </table>	TRUE	If EN is TRUE and the current cycle of the function block or function was processed without errors, ENO is also TRUE. The function cannot result in an error.	FALSE	If also EN = FALSE, or if the N input is signed and negative.
TRUE	If EN is TRUE and the current cycle of the function block or function was processed without errors, ENO is also TRUE. The function cannot result in an error.					
FALSE	If also EN = FALSE, or if the N input is signed and negative.					

3.7 SHR

This function shifts the bits of the value at IN n times to the right and fills in the bits to the left with zeros. If IN is signed and negative, the shifting direction is inverted.

3.7.1 Inputs

Input	Data type	Description		
IN	ANY_BIT	Input IN		
N	ANY_INT	Number of shifting right steps		
EN	BOOL	<p>EN is hidden by default. If EN is connected, it is always TRUE.</p> <p>Show EN/ENO: Right-click a function block or function, and select Show EN/ENO from the context menu.</p> <p>As soon as EN or ENO is connected, it can no longer be hidden.</p>		
		<table border="1"> <tr> <td>TRUE</td><td> <p>The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle.</p> <p>ENO Output: The output is TRUE and can be set to FALSE separately based on the internal error detection.</p> </td></tr> <tr> <td>FALSE</td><td> <p>The function block or function is not processed by the user program. The following variables and value field assignment are not assigned.</p> <p>The outputs of a function block display the results of the user program cycle that was lastly processed.</p> <p>The outputs of a function display the default values defined for the data type.</p> <p>As long as the function block or function has not been completely processed, the outputs retain the default values defined for the data type.</p> <p>ENO Output: The output is also set to FALSE.</p> </td></tr> </table>	TRUE	<p>The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle.</p> <p>ENO Output: The output is TRUE and can be set to FALSE separately based on the internal error detection.</p>
TRUE	<p>The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle.</p> <p>ENO Output: The output is TRUE and can be set to FALSE separately based on the internal error detection.</p>			
FALSE	<p>The function block or function is not processed by the user program. The following variables and value field assignment are not assigned.</p> <p>The outputs of a function block display the results of the user program cycle that was lastly processed.</p> <p>The outputs of a function display the default values defined for the data type.</p> <p>As long as the function block or function has not been completely processed, the outputs retain the default values defined for the data type.</p> <p>ENO Output: The output is also set to FALSE.</p>			

3.7.2 Outputs

Output	Data type	Description				
OUT	ANY_BIT	<p>The IN value is n times bitwise shifted to the right.</p> <p>The most significant bit is set to 0 after each shifting right step.</p>				
ENO	BOOL	<table border="1"> <tr> <td>TRUE</td><td>If EN is TRUE and the current cycle of the function block or function was processed without errors, ENO is also TRUE. The function cannot result in an error.</td></tr> <tr> <td>FALSE</td><td>If also EN = FALSE, or if the N input is signed and negative.</td></tr> </table>	TRUE	If EN is TRUE and the current cycle of the function block or function was processed without errors, ENO is also TRUE. The function cannot result in an error.	FALSE	If also EN = FALSE, or if the N input is signed and negative.
TRUE	If EN is TRUE and the current cycle of the function block or function was processed without errors, ENO is also TRUE. The function cannot result in an error.					
FALSE	If also EN = FALSE, or if the N input is signed and negative.					

3.8 XOR

This function performs a logical exclusive OR operation on the BOOL data type, and, with the data types BYTE through LWORD, it performs a bitwise exclusive OR operation on the inputs.

3.8.1 Inputs

Input	Data type	Description				
IN1	ANY_BIT	Input IN1				
IN2	ANY_BIT	Input IN2				
Extendible	ANY_BIT	Up to IN16				
<u>EN</u>	BOOL	<p>EN is hidden by default. If EN is connected, it is always TRUE.</p> <p>Show EN/ENO: Right-click a function block or function, and select Show EN/ENO from the context menu.</p> <p>As soon as EN or ENO is connected, it can no longer be hidden.</p> <table border="1"> <tr> <td>TRUE</td><td>The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle. ENO Output: The output is TRUE and can be set to FALSE separately based on the internal error detection.</td></tr> <tr> <td>FALSE</td><td>The function block or function is not processed by the user program. The following variables and value field assignment are not assigned. The outputs of a function block display the results of the user program cycle that was lastly processed. The outputs of a function display the default values defined for the data type. As long as the function block or function has not been completely processed, the outputs retain the default values defined for the data type. ENO Output: The output is also set to FALSE.</td></tr> </table>	TRUE	The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle. ENO Output: The output is TRUE and can be set to FALSE separately based on the internal error detection.	FALSE	The function block or function is not processed by the user program. The following variables and value field assignment are not assigned. The outputs of a function block display the results of the user program cycle that was lastly processed. The outputs of a function display the default values defined for the data type. As long as the function block or function has not been completely processed, the outputs retain the default values defined for the data type. ENO Output: The output is also set to FALSE.
TRUE	The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle. ENO Output: The output is TRUE and can be set to FALSE separately based on the internal error detection.					
FALSE	The function block or function is not processed by the user program. The following variables and value field assignment are not assigned. The outputs of a function block display the results of the user program cycle that was lastly processed. The outputs of a function display the default values defined for the data type. As long as the function block or function has not been completely processed, the outputs retain the default values defined for the data type. ENO Output: The output is also set to FALSE.					

3.8.2 Outputs

Output	Data type	Description				
OUT	ANY_BIT	<p>Performs a logical exclusive OR operation on the BOOL data type, and a bitwise exclusive OR operation on the data types BYTE to LWORD at the inputs IN1 through INn.</p> <p>OUT = IN1 XOR IN2 XOR...XOR INn</p> <p>The OUT output is 1 whenever an odd number of inputs is 1.</p>				
ENO	BOOL	<table border="1"> <tr> <td>TRUE</td><td>If EN is TRUE and the current cycle of the function block or function was processed without errors, ENO is also TRUE. The function cannot result in an error.</td></tr> <tr> <td>FALSE</td><td>Only if EN is also FALSE or while the function block or function is being processed.</td></tr> </table>	TRUE	If EN is TRUE and the current cycle of the function block or function was processed without errors, ENO is also TRUE. The function cannot result in an error.	FALSE	Only if EN is also FALSE or while the function block or function is being processed.
TRUE	If EN is TRUE and the current cycle of the function block or function was processed without errors, ENO is also TRUE. The function cannot result in an error.					
FALSE	Only if EN is also FALSE or while the function block or function is being processed.					

3.8.3 Implementation in Structured Text (ST)

OUT := IN1 XOR IN2 XOR .. XOR INn;

3.8.4 Truth table for XOR with two inputs

IN1	IN2	OUT
0	0	0
0	1	1
1	0	1
1	1	0

3.8.5 Truth table for XOR with four inputs

IN1	IN2	IN3	IN4	OUT
0	0	0	0	0
0	0	0	1	1
0	0	1	0	1
0	0	1	1	0
0	1	0	0	1
0	1	0	1	0
0	1	1	0	0
0	1	1	1	1
1	0	0	0	1
1	0	0	1	0
1	0	1	0	0
1	0	1	1	1
1	1	0	0	0
1	1	0	1	1
1	1	1	0	1
1	1	1	1	0

4 COMPARE Sub-Library

4.1 ANY_REAL Behavior

The results of comparison functions on operands of type ANY_REAL include the full width of the data type (REAL: 32 bits, LREAL: 64 bits).

With the exception of [NE](#), the output value of comparison functions on operands of type ANY_REAL is FALSE, if at least one of the input values is NaN. This also applies if all the input values are NaN.

This means that all the ANY_REAL values except for NaN can be used for comparison, even the exception values +INF and -INF.

4.2 EQ (Equal)

This function compares the values of all the IN inputs and returns FALSE if at least one value is different.

4.2.1 Inputs

Input	Data type	Description				
IN1	ANY_ELEMENTARY	Input IN1				
IN2	ANY_ELEMENTARY	Input IN2				
Extendible	ANY_ELEMENTARY	to input IN16				
EN	BOOL	<p>EN is hidden by default. If EN is connected, it is always TRUE.</p> <p>Show EN/ENO: Right-click a function block or function, and select Show EN/ENO from the context menu.</p> <p>As soon as EN or ENO is connected, it can no longer be hidden.</p> <table border="1"> <tr> <td>TRUE</td><td> The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle. ENO Output: The output is TRUE and can be set to FALSE separately based on the internal error detection. </td></tr> <tr> <td>FALSE</td><td> The function block or function is not processed by the user program. The following variables and value field assignment are not assigned. The outputs of a function block display the results of the user program cycle that was lastly processed. The outputs of a function display the default values defined for the data type. As long as the function block or function has not been completely processed, the outputs retain the default values defined for the data type. ENO Output: The output is also set to FALSE. </td></tr> </table>	TRUE	The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle. ENO Output: The output is TRUE and can be set to FALSE separately based on the internal error detection.	FALSE	The function block or function is not processed by the user program. The following variables and value field assignment are not assigned. The outputs of a function block display the results of the user program cycle that was lastly processed. The outputs of a function display the default values defined for the data type. As long as the function block or function has not been completely processed, the outputs retain the default values defined for the data type. ENO Output: The output is also set to FALSE.
TRUE	The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle. ENO Output: The output is TRUE and can be set to FALSE separately based on the internal error detection.					
FALSE	The function block or function is not processed by the user program. The following variables and value field assignment are not assigned. The outputs of a function block display the results of the user program cycle that was lastly processed. The outputs of a function display the default values defined for the data type. As long as the function block or function has not been completely processed, the outputs retain the default values defined for the data type. ENO Output: The output is also set to FALSE.					

4.2.2 Outputs

Output	Data type	Description	
OUT	BOOL	TRUE	If IN1 = IN2 = IN3 ... = INn
		FALSE	If at least two input values are different.
<u>ENO</u>	BOOL	TRUE	If EN is TRUE and the current cycle of the function block or function was processed without errors, ENO is also TRUE. The function cannot result in an error.
		FALSE	Only if EN is also FALSE or while the function block or function is being processed.

4.2.3 Implementation in Structured Text (ST)

```
OUT := IN1 = IN2 AND IN2 = IN3 AND ... AND INn-1 = INn;
```

4.3 GE (Greater or Equal)

This function compares the values of all the IN inputs and returns FALSE if at least one value is less than the value of the next input.

4.3.1 Inputs

Input	Data type	Description				
IN1	ANY_ELEMENTARY	Input IN1				
IN2	ANY_ELEMENTARY	Input IN2				
Extendible	ANY_ELEMENTARY	Up to IN16				
EN	BOOL	<p>EN is hidden by default. If EN is connected, it is always TRUE. Show EN/ENO: Right-click a function block or function, and select Show EN/ENO from the context menu.</p> <p>As soon as EN or ENO is connected, it can no longer be hidden.</p> <table border="1"> <tr> <td>TRUE</td><td> <p>The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle.</p> <p>ENO Output: The output is TRUE and can be set to FALSE separately based on the internal error detection.</p> </td></tr> <tr> <td>FALSE</td><td> <p>The function block or function is not processed by the user program. The following variables and value field assignment are not assigned.</p> <p>The outputs of a function block display the results of the user program cycle that was lastly processed.</p> <p>The outputs of a function display the default values defined for the data type.</p> <p>As long as the function block or function has not been completely processed, the outputs retain the default values defined for the data type.</p> <p>ENO Output: The output is also set to FALSE.</p> </td></tr> </table>	TRUE	<p>The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle.</p> <p>ENO Output: The output is TRUE and can be set to FALSE separately based on the internal error detection.</p>	FALSE	<p>The function block or function is not processed by the user program. The following variables and value field assignment are not assigned.</p> <p>The outputs of a function block display the results of the user program cycle that was lastly processed.</p> <p>The outputs of a function display the default values defined for the data type.</p> <p>As long as the function block or function has not been completely processed, the outputs retain the default values defined for the data type.</p> <p>ENO Output: The output is also set to FALSE.</p>
TRUE	<p>The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle.</p> <p>ENO Output: The output is TRUE and can be set to FALSE separately based on the internal error detection.</p>					
FALSE	<p>The function block or function is not processed by the user program. The following variables and value field assignment are not assigned.</p> <p>The outputs of a function block display the results of the user program cycle that was lastly processed.</p> <p>The outputs of a function display the default values defined for the data type.</p> <p>As long as the function block or function has not been completely processed, the outputs retain the default values defined for the data type.</p> <p>ENO Output: The output is also set to FALSE.</p>					

4.3.2 Outputs

Output	Data type	Description	
OUT	BOOL	TRUE	If $IN1 \geq IN2 \geq IN3 \dots \geq INn$
		FALSE	If one or more conditions are not met.
<u>ENO</u>	BOOL	TRUE	If EN is TRUE and the current cycle of the function block or function was processed without errors, ENO is also TRUE. The function cannot result in an error.
		FALSE	Only if EN is also FALSE or while the function block or function is being processed.

4.3.3 Implementation in Structured Text (ST)

```
OUT := IN1 >= IN2 AND IN2 >= IN3 AND .. AND INn-1 >= INn;
```

4.4 GT (Greater Than)

This function compares the values of all the IN inputs and returns FALSE if at least one value is less than or equal to the value of the next input.

4.4.1 Inputs

Input	Data type	Description				
IN1	ANY ELEMENTARY	Input IN1				
IN2	ANY ELEMENTARY	Input IN2				
Extendible	ANY ELEMENTARY	Up to IN16				
EN	BOOL	<p>EN is hidden by default. If EN is connected, it is always TRUE. Show EN/ENO: Right-click a function block or function, and select Show EN/ENO from the context menu.</p> <p>As soon as EN or ENO is connected, it can no longer be hidden.</p> <table border="1"> <tr> <td>TRUE</td><td>The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle. ENO Output: The output is TRUE and can be set to FALSE separately based on the internal error detection.</td></tr> <tr> <td>FALSE</td><td>The function block or function is not processed by the user program. The following variables and value field assignment are not assigned. The outputs of a function block display the results of the user program cycle that was lastly processed. The outputs of a function display the default values defined for the data type. As long as the function block or function has not been completely processed, the outputs retain the default values defined for the data type. ENO Output: The output is also set to FALSE.</td></tr> </table>	TRUE	The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle. ENO Output: The output is TRUE and can be set to FALSE separately based on the internal error detection.	FALSE	The function block or function is not processed by the user program. The following variables and value field assignment are not assigned. The outputs of a function block display the results of the user program cycle that was lastly processed. The outputs of a function display the default values defined for the data type. As long as the function block or function has not been completely processed, the outputs retain the default values defined for the data type. ENO Output: The output is also set to FALSE.
TRUE	The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle. ENO Output: The output is TRUE and can be set to FALSE separately based on the internal error detection.					
FALSE	The function block or function is not processed by the user program. The following variables and value field assignment are not assigned. The outputs of a function block display the results of the user program cycle that was lastly processed. The outputs of a function display the default values defined for the data type. As long as the function block or function has not been completely processed, the outputs retain the default values defined for the data type. ENO Output: The output is also set to FALSE.					

4.4.2 Outputs

Output	Data type	Description	
OUT	BOOL	TRUE	If IN1 > IN2 > IN3...> INn
		FALSE	If one or more conditions are not met.
<u>ENO</u>	BOOL	TRUE	If EN is TRUE and the current cycle of the function block or function was processed without errors, ENO is also TRUE. The function cannot result in an error.
		FALSE	Only if EN is also FALSE or while the function block or function is being processed.

4.4.3 Implementation in Structured Text (ST)

```
OUT := IN1 > IN2 AND IN2 > IN3 AND ... AND INn-1 > INn;
```

4.5 LE (Less or Equal)

This function compares the values of all the IN inputs and returns FALSE if at least one value is greater than the value of the next input.

4.5.1 Inputs

Input	Data type	Description				
IN1	ANY ELEMENTARY	Input IN1				
IN2	ANY ELEMENTARY	Input IN2				
Extendible	ANY ELEMENTARY	Up to IN16				
EN	BOOL	<p>EN is hidden by default. If EN is connected, it is always TRUE. Show EN/ENO: Right-click a function block or function, and select Show EN/ENO from the context menu.</p> <p>As soon as EN or ENO is connected, it can no longer be hidden.</p> <table border="1"> <tr> <td>TRUE</td><td>The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle. ENO Output: The output is TRUE and can be set to FALSE separately based on the internal error detection.</td></tr> <tr> <td>FALSE</td><td>The function block or function is not processed by the user program. The following variables and value field assignment are not assigned. The outputs of a function block display the results of the user program cycle that was lastly processed. The outputs of a function display the default values defined for the data type. As long as the function block or function has not been completely processed, the outputs retain the default values defined for the data type. ENO Output: The output is also set to FALSE.</td></tr> </table>	TRUE	The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle. ENO Output: The output is TRUE and can be set to FALSE separately based on the internal error detection.	FALSE	The function block or function is not processed by the user program. The following variables and value field assignment are not assigned. The outputs of a function block display the results of the user program cycle that was lastly processed. The outputs of a function display the default values defined for the data type. As long as the function block or function has not been completely processed, the outputs retain the default values defined for the data type. ENO Output: The output is also set to FALSE.
TRUE	The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle. ENO Output: The output is TRUE and can be set to FALSE separately based on the internal error detection.					
FALSE	The function block or function is not processed by the user program. The following variables and value field assignment are not assigned. The outputs of a function block display the results of the user program cycle that was lastly processed. The outputs of a function display the default values defined for the data type. As long as the function block or function has not been completely processed, the outputs retain the default values defined for the data type. ENO Output: The output is also set to FALSE.					

4.5.2 Outputs

Output	Data type	Description	
OUT	BOOL	TRUE	If $IN1 \leq IN2 \leq IN3 \dots \leq INn$
		FALSE	If one or more conditions are not met.
<u>ENO</u>	BOOL	TRUE	If EN is TRUE and the current cycle of the function block or function was processed without errors, ENO is also TRUE. The function cannot result in an error.
		FALSE	Only if EN is also FALSE or while the function block or function is being processed.

4.5.3 Implementation in Structured Text (ST)

```
OUT := IN1 <= IN2 AND IN2 <= IN3 AND .. AND INn-1 <= INn;
```

4.6 LT (Less Than)

This function compares the values of all the IN inputs and returns FALSE if at least one value is greater than or equal to the value of the next input.

4.6.1 Inputs

Input	Data type	Description				
IN1	ANY ELEMENTARY	Input IN1				
IN2	ANY ELEMENTARY	Input IN2				
Extendible	ANY ELEMENTARY	Up to IN16				
EN	BOOL	<p>EN is hidden by default. If EN is connected, it is always TRUE. Show EN/ENO: Right-click a function block or function, and select Show EN/ENO from the context menu.</p> <p>As soon as EN or ENO is connected, it can no longer be hidden.</p> <table border="1"> <tr> <td>TRUE</td><td>The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle. ENO Output: The output is TRUE and can be set to FALSE separately based on the internal error detection.</td></tr> <tr> <td>FALSE</td><td>The function block or function is not processed by the user program. The following variables and value field assignment are not assigned. The outputs of a function block display the results of the user program cycle that was lastly processed. The outputs of a function display the default values defined for the data type. As long as the function block or function has not been completely processed, the outputs retain the default values defined for the data type. ENO Output: The output is also set to FALSE.</td></tr> </table>	TRUE	The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle. ENO Output: The output is TRUE and can be set to FALSE separately based on the internal error detection.	FALSE	The function block or function is not processed by the user program. The following variables and value field assignment are not assigned. The outputs of a function block display the results of the user program cycle that was lastly processed. The outputs of a function display the default values defined for the data type. As long as the function block or function has not been completely processed, the outputs retain the default values defined for the data type. ENO Output: The output is also set to FALSE.
TRUE	The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle. ENO Output: The output is TRUE and can be set to FALSE separately based on the internal error detection.					
FALSE	The function block or function is not processed by the user program. The following variables and value field assignment are not assigned. The outputs of a function block display the results of the user program cycle that was lastly processed. The outputs of a function display the default values defined for the data type. As long as the function block or function has not been completely processed, the outputs retain the default values defined for the data type. ENO Output: The output is also set to FALSE.					

4.6.2 Outputs

Output	Data type	Description	
OUT	BOOL	TRUE	If IN1 < IN2 < IN3...< INn
		FALSE	If one or more conditions are not met.
<u>ENO</u>	BOOL	TRUE	If EN is TRUE and the current cycle of the function block or function was processed without errors, ENO is also TRUE. The function cannot result in an error.
		FALSE	Only if EN is also FALSE or while the function block or function is being processed.

4.6.3 Implementation in Structured Text (ST)

```
OUT := IN1 < IN2 AND IN2 < IN3 AND ... AND INn-1 < INn;
```

4.7 NE (Not Equal)

This function compares the values of the inputs IN1 and IN2, and returns FALSE if the values are identical.

4.7.1 Behavior if the input is NaN

For variables of type REAL or LREAL, the value can be NaN. OUT is TRUE if at least one of the two input values is NaN.

Two NaN values are always considered as unequal, even if they have identical bit pattern. Whether a variable has the value NaN can be checked by comparing the variable with itself.

For a variable A with the value NaN:

- $\text{NE}(A, A) \rightarrow \text{TRUE}$
- $\text{EQ}(A, A) \rightarrow \text{FALSE}$

4.7.2 Inputs

Input	Data type	Description				
IN1	ANY ELEMENTARY	Input IN1				
IN2	ANY ELEMENTARY	Input IN2				
EN	BOOL	<p>EN is hidden by default. If EN is connected, it is always TRUE.</p> <p>Show EN/ENO: Right-click a function block or function, and select Show EN/ENO from the context menu.</p> <p>As soon as EN or ENO is connected, it can no longer be hidden.</p> <table border="1"> <tr> <td>TRUE</td><td>The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle. ENO Output: The output is TRUE and can be set to FALSE separately based on the internal error detection.</td></tr> <tr> <td>FALSE</td><td>The function block or function is not processed by the user program. The following variables and value field assignment are not assigned. The outputs of a function block display the results of the user program cycle that was lastly processed. The outputs of a function display the default values defined for the data type. As long as the function block or function has not been completely processed, the outputs retain the default values defined for the data type. ENO Output: The output is also set to FALSE.</td></tr> </table>	TRUE	The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle. ENO Output: The output is TRUE and can be set to FALSE separately based on the internal error detection.	FALSE	The function block or function is not processed by the user program. The following variables and value field assignment are not assigned. The outputs of a function block display the results of the user program cycle that was lastly processed. The outputs of a function display the default values defined for the data type. As long as the function block or function has not been completely processed, the outputs retain the default values defined for the data type. ENO Output: The output is also set to FALSE.
TRUE	The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle. ENO Output: The output is TRUE and can be set to FALSE separately based on the internal error detection.					
FALSE	The function block or function is not processed by the user program. The following variables and value field assignment are not assigned. The outputs of a function block display the results of the user program cycle that was lastly processed. The outputs of a function display the default values defined for the data type. As long as the function block or function has not been completely processed, the outputs retain the default values defined for the data type. ENO Output: The output is also set to FALSE.					

4.7.3 Outputs

Output	Data type	Description	
OUT	BOOL	TRUE	If IN1 ≠ IN2
		FALSE	If the two input values are identical.
<u>ENO</u>	BOOL	TRUE	If EN is TRUE and the current cycle of the function block or function was processed without errors, ENO is also TRUE. The function cannot result in an error.
		FALSE	Only if EN is also FALSE or while the function block or function is being processed.

4.7.4 Implementation in Structured Text (ST)

```
OUT := IN1 <> IN2;
```

5 CONVERT Sub-Library

Rounding effects must be expected whenever INT data types \geq 32 bits are converted to REAL and 64-bit ANY_INT are converted to LREAL. This is due to the binary processing of floating point numbers in microprocessors, in accordance with DIN IEC 60559:1989.

All the TRUNC* and ATO* functions except for [ATOBOOL](#) and [ATOLREAL](#) return an error to ENO if the input value is not within the range of values of the target data type. If this happens, the function returns the portion that can be represented by the range of values of the output.

5.1 ANY_REAL Behavior

If TRUNC, ANY_REAL \rightarrow ANY_INT and ANY_REAL \rightarrow ANY_BIT except for ANY_REAL \rightarrow BOOL, and ANY_REAL exception value (NaN, +INF, -INF) is present at the input, OUT is set to 0.

If the input is an exception value (NaN, +INF, -INF) when converting from ANY_REAL \rightarrow BOOL, OUT is set to TRUE.

5.2 ATOBOOL (Any to BOOL)

The OUT output is FALSE if the value 0 (0.0, T#0s) is present at the IN input. For all the other values at IN, OUT = TRUE.

5.2.1 Inputs

Input	Data type	Description				
IN	ANY	Input IN1				
EN	BOOL	<p>EN is hidden by default. If EN is connected, it is always TRUE. Show EN/ENO: Right-click a function block or function, and select Show EN/ENO from the context menu.</p> <p>As soon as EN or ENO is connected, it can no longer be hidden.</p> <table border="1"> <tr> <td>TRUE</td><td> The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle. ENO Output: The output is TRUE and can be set to FALSE separately based on the internal error detection. </td></tr> <tr> <td>FALSE</td><td> The function block or function is not processed by the user program. The following variables and value field assignment are not assigned. The outputs of a function block display the results of the user program cycle that was lastly processed. The outputs of a function display the default values defined for the data type. As long as the function block or function has not been completely processed, the outputs retain the default values defined for the data type. ENO Output: The output is also set to FALSE. </td></tr> </table>	TRUE	The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle. ENO Output : The output is TRUE and can be set to FALSE separately based on the internal error detection.	FALSE	The function block or function is not processed by the user program. The following variables and value field assignment are not assigned. The outputs of a function block display the results of the user program cycle that was lastly processed. The outputs of a function display the default values defined for the data type. As long as the function block or function has not been completely processed, the outputs retain the default values defined for the data type. ENO Output : The output is also set to FALSE.
TRUE	The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle. ENO Output : The output is TRUE and can be set to FALSE separately based on the internal error detection.					
FALSE	The function block or function is not processed by the user program. The following variables and value field assignment are not assigned. The outputs of a function block display the results of the user program cycle that was lastly processed. The outputs of a function display the default values defined for the data type. As long as the function block or function has not been completely processed, the outputs retain the default values defined for the data type. ENO Output : The output is also set to FALSE.					

5.2.2 Outputs

Output	Data type	Description	
OUT	BOOL	Converts the data type at IN1 to BOOL. The created typecasting line determines the data type at IN1.	
		TRUE	If IN1 ≠ 0 (or 0.0, T#0s).
		FALSE	If IN1 = 0 (or 0.0, T#0s).
		If the input is an exception value (NaN, +INF, -INF) when converting from ANY_REAL → BOOL, OUT is set to TRUE.	
<u>ENO</u>	BOOL	TRUE	If EN is TRUE and the current cycle of the function block or function was processed without errors, ENO is also TRUE. The function cannot result in an error.
		FALSE	Only if EN is also FALSE or while the function block or function is being processed.

5.2.3 Conversion Within the Ranges of Values

Data type at IN1	Input lower limit	Output lower limit	Input upper limit	Output upper limit
BOOL	0	0	1	1
BYTE	16#00	0	16#01	1
DINT	0	0	1	1
DWORD	16#00	0	16#01	1
INT	0	0	1	1
LREAL	0.0	0	1.0	1
REAL	0.0	0	1.0	1
SINT	0	0	1	1
TIME	T#0s	0	T#1s	1
UDINT	0	0	1	1
UINT	0	0	1	1
USINT	0	0	1	1
WORD	16#00	0	16#01	1

5.2.4 Conversion Outside the Ranges of Values

Data type at IN1	Input lower limit – 1	Output lower limit	Input upper limit + 1	Output upper limit
BOOL	-/-	-/-	-/-	-/-
BYTE	-/-	-/-	16#02	1
DINT	-1	1	2	1
DWORD	-/-	-/-	16#02	1
INT	-1	1	2	1
LREAL	-1.0	1	2.0	1
REAL	-1.0	1	2.0	1
SINT	-1	1	2	0
TIME	T#-1s	1	T#2s	1
UDINT	-/-	-/-	2	1
UINT	-/-	-/-	2	1
USINT	-/-	-/-	2	1
WORD	-/-	-/-	16#02	1

5.3 ATOBYTE (Any to BYTE)

5.3.1 Inputs

Input	Data type	Description				
IN	ANY ELEMENTARY	<p>Input IN</p>				
EN	BOOL	<p>EN is hidden by default. If EN is connected, it is always TRUE.</p> <p>Show EN/ENO: Right-click a function block or function, and select Show EN/ENO from the context menu.</p> <p>As soon as EN or ENO is connected, it can no longer be hidden.</p> <table border="1"> <tr> <td>TRUE</td><td> <p>The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle.</p> <p>ENO Output: The output is TRUE and can be set to FALSE separately based on the internal error detection.</p> </td></tr> <tr> <td>FALSE</td><td> <p>The function block or function is not processed by the user program. The following variables and value field assignment are not assigned.</p> <p>The outputs of a function block display the results of the user program cycle that was lastly processed.</p> <p>The outputs of a function display the default values defined for the data type.</p> <p>As long as the function block or function has not been completely processed, the outputs retain the default values defined for the data type.</p> <p>ENO Output: The output is also set to FALSE.</p> </td></tr> </table>	TRUE	<p>The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle.</p> <p>ENO Output: The output is TRUE and can be set to FALSE separately based on the internal error detection.</p>	FALSE	<p>The function block or function is not processed by the user program. The following variables and value field assignment are not assigned.</p> <p>The outputs of a function block display the results of the user program cycle that was lastly processed.</p> <p>The outputs of a function display the default values defined for the data type.</p> <p>As long as the function block or function has not been completely processed, the outputs retain the default values defined for the data type.</p> <p>ENO Output: The output is also set to FALSE.</p>
TRUE	<p>The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle.</p> <p>ENO Output: The output is TRUE and can be set to FALSE separately based on the internal error detection.</p>					
FALSE	<p>The function block or function is not processed by the user program. The following variables and value field assignment are not assigned.</p> <p>The outputs of a function block display the results of the user program cycle that was lastly processed.</p> <p>The outputs of a function display the default values defined for the data type.</p> <p>As long as the function block or function has not been completely processed, the outputs retain the default values defined for the data type.</p> <p>ENO Output: The output is also set to FALSE.</p>					

5.3.2 Outputs

Output	Data type	Description																																																						
		<p>Converts the data type at IN to BYTE.</p> <p>The following conditions apply to the value of OUT.</p> <p>Input data type: BOOL</p> <table border="1"> <thead> <tr> <th>Description</th><th colspan="2">Examples</th></tr> <tr> <th></th><th>IN</th><th>OUT</th></tr> </thead> <tbody> <tr> <td>The value TRUE and FALSE at IN are output to OUT as 1 and 0, respectively.</td><td>TRUE</td><td>1</td></tr> <tr> <td></td><td>FALSE</td><td>0</td></tr> </tbody> </table> <p>Input data type: ANY_INT</p> <table border="1"> <thead> <tr> <th>Description</th><th colspan="2">Examples</th></tr> <tr> <th></th><th>IN</th><th>OUT</th></tr> </thead> <tbody> <tr> <td>If the value of IN exceeds the range of values defined for BYTE, the portion that can be represented with the BYTE data type is output to OUT, and ENO is set to FALSE. If IN is negative, the corresponding two's complement is output to OUT.</td><td>257 (16#101)</td><td>01</td></tr> <tr> <td></td><td>12FF</td><td>FF</td></tr> <tr> <td></td><td>-2</td><td>254 (16#FE)</td></tr> </tbody> </table> <p>Input data type: ANY_REAL</p> <table border="1"> <thead> <tr> <th>Description</th><th colspan="2">Examples</th></tr> <tr> <th></th><th>IN</th><th>OUT</th></tr> </thead> <tbody> <tr> <td>In accordance with DIN 1333, REAL data types are rounded. If the first decimal place is equal to or greater than 5, the data type is rounded up, otherwise it is rounded down. If IN is negative, the corresponding two's complement is output to OUT.</td><td>2.5</td><td>3</td></tr> <tr> <td></td><td>1.4</td><td>1</td></tr> <tr> <td></td><td>-2.5</td><td>253 (16#FD)</td></tr> <tr> <td></td><td>-1.4</td><td>255 (16#FF)</td></tr> <tr> <td></td><td>-8.4</td><td>248 (16#F8)</td></tr> <tr> <td>If the modulus of the rounded value at IN is greater than the value that can be represented with the BYTE data type, but less than the largest unsigned 32-bit integer value, an overflow occurs, and ENO is set to FALSE.</td><td>257.0</td><td>1 (16#01)</td></tr> <tr> <td>If the modulus of the rounded value at IN is greater than the largest unsigned 32-bit integer value, the largest value that can be represented with the BYTE data type is output to OUT, and ENO is set to FALSE.</td><td>3.66455 e+18</td><td>255 (16#FF)</td></tr> </tbody> </table>	Description	Examples			IN	OUT	The value TRUE and FALSE at IN are output to OUT as 1 and 0, respectively.	TRUE	1		FALSE	0	Description	Examples			IN	OUT	If the value of IN exceeds the range of values defined for BYTE, the portion that can be represented with the BYTE data type is output to OUT, and ENO is set to FALSE. If IN is negative, the corresponding two's complement is output to OUT.	257 (16#101)	01		12FF	FF		-2	254 (16#FE)	Description	Examples			IN	OUT	In accordance with DIN 1333, REAL data types are rounded. If the first decimal place is equal to or greater than 5, the data type is rounded up, otherwise it is rounded down. If IN is negative, the corresponding two's complement is output to OUT.	2.5	3		1.4	1		-2.5	253 (16#FD)		-1.4	255 (16#FF)		-8.4	248 (16#F8)	If the modulus of the rounded value at IN is greater than the value that can be represented with the BYTE data type, but less than the largest unsigned 32-bit integer value, an overflow occurs, and ENO is set to FALSE.	257.0	1 (16#01)	If the modulus of the rounded value at IN is greater than the largest unsigned 32-bit integer value, the largest value that can be represented with the BYTE data type is output to OUT, and ENO is set to FALSE.	3.66455 e+18	255 (16#FF)
Description	Examples																																																							
	IN	OUT																																																						
The value TRUE and FALSE at IN are output to OUT as 1 and 0, respectively.	TRUE	1																																																						
	FALSE	0																																																						
Description	Examples																																																							
	IN	OUT																																																						
If the value of IN exceeds the range of values defined for BYTE, the portion that can be represented with the BYTE data type is output to OUT, and ENO is set to FALSE. If IN is negative, the corresponding two's complement is output to OUT.	257 (16#101)	01																																																						
	12FF	FF																																																						
	-2	254 (16#FE)																																																						
Description	Examples																																																							
	IN	OUT																																																						
In accordance with DIN 1333, REAL data types are rounded. If the first decimal place is equal to or greater than 5, the data type is rounded up, otherwise it is rounded down. If IN is negative, the corresponding two's complement is output to OUT.	2.5	3																																																						
	1.4	1																																																						
	-2.5	253 (16#FD)																																																						
	-1.4	255 (16#FF)																																																						
	-8.4	248 (16#F8)																																																						
If the modulus of the rounded value at IN is greater than the value that can be represented with the BYTE data type, but less than the largest unsigned 32-bit integer value, an overflow occurs, and ENO is set to FALSE.	257.0	1 (16#01)																																																						
If the modulus of the rounded value at IN is greater than the largest unsigned 32-bit integer value, the largest value that can be represented with the BYTE data type is output to OUT, and ENO is set to FALSE.	3.66455 e+18	255 (16#FF)																																																						

Output	Data type	Description					
OUT	BYTE	Input data type: TIME					
		Description	Examples				
		If the IN data type is TIME, the value at IN is interpreted in milliseconds [ms] and converted to BYTE.	<table border="1" style="margin-left: auto; margin-right: auto;"> <tr> <td>IN</td><td>OUT</td></tr> <tr> <td>T#255ms</td><td>255 (16#FF)</td></tr> </table>	IN	OUT	T#255ms	255 (16#FF)
IN	OUT						
T#255ms	255 (16#FF)						
<u>ENO</u>	BOOL	<p>TRUE If EN = TRUE and the function was processed without errors.</p> <p>FALSE • If also EN = FALSE or • Error: The result is out of the range of values for the BYTE data type.</p>					

Only the data types specified in the table below can be converted.

For more details on the range of values, refer to [Converting LREAL to ANYBIT and ANYINT](#).

5.3.3 Conversion Within the Ranges of Values

Data type at IN	Lower limit	Output lower limit	Input upper limit	Output upper limit
BOOL	0	16#00	1	16#01
BYTE	16#00	16#00	16#FF	16#FF
DINT	0	16#00	255	16#FF
DWORD	16#00	16#00	16#FF	16#FF
INT	0	16#00	255	16#FF
LREAL	0.0	16#00	255.0	16#FF
REAL	0.0	16#00	255.0	16#FF
SINT	0	16#00	127	16#7F
TIME	T#0ms	16#00	T#255ms	16#FF
UDINT	0	16#00	255	16#FF
UINT	0	16#00	255	16#FF
USINT	0	16#00	255	16#FF
WORD	16#00	16#00	16#FF	16#FF

5.3.4 Conversion Outside the Ranges of Values

Data type at IN	Input lower limit – 1	Output lower limit	Input upper limit + 1	Output upper limit
BOOL	-/-	-/-	-/-	-/-
BYTE	-/-	-/-	-/-	-/-
DINT	-1	16#FF	256	16#00
DWORD	-/-	-/-	16#0100	16#00
INT	-1	16#FF	256	16#00
LREAL	-1.0	16#FF	256.0	16#00
REAL	-1.0	16#FF	256.0	16#00
SINT	-1	16#FF	-/-	-/-
TIME	T#-1ms	16#00	T#256ms	16#00
UDINT	-/-	-/-	256	16#00
UINT	-/-	-/-	256	16#00
USINT	-/-	-/-	-/-	-/-
WORD	-/-	-/-	16#0100	16#00

5.3.5 Conversion of LREAL to ANYBIT and ANYINT

5.3.5.1 LREAL_TO_ANYBIT

LREAL value	Data type	BYTE	WORD	DWORD	LWORD
Max	16#ff	16#ffff	16#ffff_ffff	16#ffffffff_fffffff	
Min	16#00	16#0000	16#0000_0000	16#00000000_00000000	
-1.0	16#ff	16#ffff	16#ffff_ffff	16#ffffffff_fffffff	
-2.0	16#fe	16#ffffe	16#ffff_ffffe	16#fffffff_ffffffe	
256.0	16#00	16#0100	16#0000_0100	16#00000000_00000100	
-2147483647.0	16#01	16#0001	16#8000_0001	16#ffffffff_80000001	
2147483647.0	16#ff	16#ffff	16#7fff_ffff	16#00000000_7fffffff	
-4294967295.0	16#00	16#0000	16#8000_0000	16#ffffffff_00000001	
4294967295.0	16#ff	16#ffff	16#ffff_ffff	16#00000000_ffffffff	
-4294967296.0	16#00	16#0000	16#8000_0000	16#ffffffff_00000000	
4294967296.0	16#ff	16#ffff	16#ffff_ffff	16#00000001_00000000	
-1.79769313486157e+308	16#00	16#0000	16#8000_0000	16#80000000_00000000	
1.79769313486157e+308	16#ff	16#ffff	16#ffff_ffff	16#ffffffff_fffffff	

5.3.5.2 LREAL_TO_ANYINT

LREAL value	Data type	USINT	SINT	UINT	INT	UDINT	DINT
	Max	255	127	65535	32767	4294967295	2147483647
	Min	0	-128	0	-32768	0	-2147483648
-1.0	255	-1	65535	-1	4294967295	-1	
-2.0	254	-2	65534	-2	4294967294	-2	
-2147483647.0	1	1	1	1	2147483649	-2147483647	
2147483647.0	255	-1	65535	-1	2147483647	2147483647	
-4294967295.0	0	0	0	0	2147483648	-2147483648	
4294967295.0	255	-1	65535	-1	4294967295	2147483647	
-1.79769313486157e+308	0	0	0	0	2147483648	-2147483648	
1.79769313486157e+308	255	-1	65535	-1	4294967295	2147483647	

LREAL value	Data type	ULINT	LINT
	Max	18446744073709551615	9223372036854775807
	Min	0	-9223372036854775808
-1.0	18446744073709551615	-1	
-2.0	18446744073709551614	-2	
-2147483647.0	18446744071562067969	-2147483647	
2147483647.0	2147483647	2147483647	
-4294967295.0	18446744069414584321	-4294967295	
4294967295.0	4294967295	4294967295	
-1.79769313486157e+308	9223372036854775808	-9223372036854775808	
1.79769313486157e+308	18446744073709551615	9223372036854775807	

5.4 ATOWORD (Any to WORD)

5.4.1 Inputs

Input	Data type	Description				
IN	ANY ELEMENTARY	Input IN				
<u>EN</u>	BOOL	<p>EN is hidden by default. If EN is connected, it is always TRUE.</p> <p>Show EN/ENO: Right-click a function block or function, and select Show EN/ENO from the context menu.</p> <p>As soon as EN or ENO is connected, it can no longer be hidden.</p> <table border="1"> <tr> <td>TRUE</td><td>The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle. ENO Output: The output is TRUE and can be set to FALSE separately based on the internal error detection.</td></tr> <tr> <td>FALSE</td><td>The function block or function is not processed by the user program. The following variables and value field assignment are not assigned. The outputs of a function block display the results of the user program cycle that was lastly processed. The outputs of a function display the default values defined for the data type. As long as the function block or function has not been completely processed, the outputs retain the default values defined for the data type. ENO Output: The output is also set to FALSE.</td></tr> </table>	TRUE	The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle. ENO Output: The output is TRUE and can be set to FALSE separately based on the internal error detection.	FALSE	The function block or function is not processed by the user program. The following variables and value field assignment are not assigned. The outputs of a function block display the results of the user program cycle that was lastly processed. The outputs of a function display the default values defined for the data type. As long as the function block or function has not been completely processed, the outputs retain the default values defined for the data type. ENO Output: The output is also set to FALSE.
TRUE	The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle. ENO Output: The output is TRUE and can be set to FALSE separately based on the internal error detection.					
FALSE	The function block or function is not processed by the user program. The following variables and value field assignment are not assigned. The outputs of a function block display the results of the user program cycle that was lastly processed. The outputs of a function display the default values defined for the data type. As long as the function block or function has not been completely processed, the outputs retain the default values defined for the data type. ENO Output: The output is also set to FALSE.					

5.4.2 Outputs

Output	Data type	Description																																																
		<p>Converts the data type at IN to WORD.</p> <p>The data type at IN is determined by creating a typecasting line.</p> <p>Input data type: BOOL</p> <table border="1"> <thead> <tr> <th>Description</th><th>Examples</th></tr> </thead> <tbody> <tr> <td>The value TRUE and FALSE at IN are output to OUT as 1 and 0, respectively.</td><td> <table border="1"> <thead> <tr> <th>IN</th><th>OUT</th></tr> </thead> <tbody> <tr> <td>TRUE</td><td>16#0001</td></tr> <tr> <td>FALSE</td><td>16#0000</td></tr> </tbody> </table> </td></tr> </tbody> </table> <p>Input data type: ANY_INT</p> <table border="1"> <thead> <tr> <th>Description</th><th>Examples</th></tr> </thead> <tbody> <tr> <td>If IN is greater than the value that can be represented with WORD, but less than $2^{32} - 1$, IN is limited to [IN modulus 65536] and ENO is FALSE. Whenever the value of IN is greater than or equal to 2^{32}, FFFF is output and ENO is FALSE. If IN is negative, the corresponding two's complement is output to OUT and ENO is FALSE.</td><td> <table border="1"> <thead> <tr> <th>IN</th><th>OUT</th></tr> </thead> <tbody> <tr> <td>$2^{16} - 1$</td><td>16#FFFF</td></tr> <tr> <td>$2^{32} + 1$</td><td>16#FFFF</td></tr> <tr> <td>-1</td><td>16#FFFF</td></tr> <tr> <td>-2</td><td>16#FFFE</td></tr> </tbody> </table> </td></tr> </tbody> </table> <p>Input data type: ANY_REAL</p> <table border="1"> <thead> <tr> <th>Description</th><th>Examples</th></tr> </thead> <tbody> <tr> <td>In accordance with DIN 1333, REAL data types are rounded. If the first decimal place is equal to or greater than 5, the data type is rounded up, otherwise it is rounded down. If IN is negative, the corresponding two's complement is output to OUT and ENO is FALSE.</td><td> <table border="1"> <thead> <tr> <th>IN</th><th>OUT</th></tr> </thead> <tbody> <tr> <td>2.5</td><td>16#0003</td></tr> <tr> <td>1.4</td><td>16#0001</td></tr> <tr> <td>-2.5</td><td>16#FFFD</td></tr> <tr> <td>-1.4</td><td>16#FFFF</td></tr> <tr> <td>-8.4</td><td>16#FFF8</td></tr> </tbody> </table> </td></tr> <tr> <td>If the modulus of the rounded value of IN is greater than the value that can be represented with WORD, but less than $2^{32} - 1$, IN is limited to [IN modulus 65536.0] and ENO is FALSE. If the modulus of the rounded value of IN is greater than or equal to 2^{32} FFFF is output and ENO is FALSE.</td><td> <table border="1"> <thead> <tr> <th>IN</th><th>OUT</th></tr> </thead> <tbody> <tr> <td>97680.0</td><td>16#7D90</td></tr> <tr> <td>1.1e+23</td><td>16#FFFF</td></tr> </tbody> </table> </td></tr> </tbody> </table>	Description	Examples	The value TRUE and FALSE at IN are output to OUT as 1 and 0, respectively.	<table border="1"> <thead> <tr> <th>IN</th><th>OUT</th></tr> </thead> <tbody> <tr> <td>TRUE</td><td>16#0001</td></tr> <tr> <td>FALSE</td><td>16#0000</td></tr> </tbody> </table>	IN	OUT	TRUE	16#0001	FALSE	16#0000	Description	Examples	If IN is greater than the value that can be represented with WORD, but less than $2^{32} - 1$, IN is limited to [IN modulus 65536] and ENO is FALSE. Whenever the value of IN is greater than or equal to 2^{32} , FFFF is output and ENO is FALSE. If IN is negative, the corresponding two's complement is output to OUT and ENO is FALSE.	<table border="1"> <thead> <tr> <th>IN</th><th>OUT</th></tr> </thead> <tbody> <tr> <td>$2^{16} - 1$</td><td>16#FFFF</td></tr> <tr> <td>$2^{32} + 1$</td><td>16#FFFF</td></tr> <tr> <td>-1</td><td>16#FFFF</td></tr> <tr> <td>-2</td><td>16#FFFE</td></tr> </tbody> </table>	IN	OUT	$2^{16} - 1$	16#FFFF	$2^{32} + 1$	16#FFFF	-1	16#FFFF	-2	16#FFFE	Description	Examples	In accordance with DIN 1333, REAL data types are rounded. If the first decimal place is equal to or greater than 5, the data type is rounded up, otherwise it is rounded down. If IN is negative, the corresponding two's complement is output to OUT and ENO is FALSE.	<table border="1"> <thead> <tr> <th>IN</th><th>OUT</th></tr> </thead> <tbody> <tr> <td>2.5</td><td>16#0003</td></tr> <tr> <td>1.4</td><td>16#0001</td></tr> <tr> <td>-2.5</td><td>16#FFFD</td></tr> <tr> <td>-1.4</td><td>16#FFFF</td></tr> <tr> <td>-8.4</td><td>16#FFF8</td></tr> </tbody> </table>	IN	OUT	2.5	16#0003	1.4	16#0001	-2.5	16#FFFD	-1.4	16#FFFF	-8.4	16#FFF8	If the modulus of the rounded value of IN is greater than the value that can be represented with WORD, but less than $2^{32} - 1$, IN is limited to [IN modulus 65536.0] and ENO is FALSE. If the modulus of the rounded value of IN is greater than or equal to 2^{32} FFFF is output and ENO is FALSE.	<table border="1"> <thead> <tr> <th>IN</th><th>OUT</th></tr> </thead> <tbody> <tr> <td>97680.0</td><td>16#7D90</td></tr> <tr> <td>1.1e+23</td><td>16#FFFF</td></tr> </tbody> </table>	IN	OUT	97680.0	16#7D90	1.1e+23	16#FFFF
Description	Examples																																																	
The value TRUE and FALSE at IN are output to OUT as 1 and 0, respectively.	<table border="1"> <thead> <tr> <th>IN</th><th>OUT</th></tr> </thead> <tbody> <tr> <td>TRUE</td><td>16#0001</td></tr> <tr> <td>FALSE</td><td>16#0000</td></tr> </tbody> </table>	IN	OUT	TRUE	16#0001	FALSE	16#0000																																											
IN	OUT																																																	
TRUE	16#0001																																																	
FALSE	16#0000																																																	
Description	Examples																																																	
If IN is greater than the value that can be represented with WORD, but less than $2^{32} - 1$, IN is limited to [IN modulus 65536] and ENO is FALSE. Whenever the value of IN is greater than or equal to 2^{32} , FFFF is output and ENO is FALSE. If IN is negative, the corresponding two's complement is output to OUT and ENO is FALSE.	<table border="1"> <thead> <tr> <th>IN</th><th>OUT</th></tr> </thead> <tbody> <tr> <td>$2^{16} - 1$</td><td>16#FFFF</td></tr> <tr> <td>$2^{32} + 1$</td><td>16#FFFF</td></tr> <tr> <td>-1</td><td>16#FFFF</td></tr> <tr> <td>-2</td><td>16#FFFE</td></tr> </tbody> </table>	IN	OUT	$2^{16} - 1$	16#FFFF	$2^{32} + 1$	16#FFFF	-1	16#FFFF	-2	16#FFFE																																							
IN	OUT																																																	
$2^{16} - 1$	16#FFFF																																																	
$2^{32} + 1$	16#FFFF																																																	
-1	16#FFFF																																																	
-2	16#FFFE																																																	
Description	Examples																																																	
In accordance with DIN 1333, REAL data types are rounded. If the first decimal place is equal to or greater than 5, the data type is rounded up, otherwise it is rounded down. If IN is negative, the corresponding two's complement is output to OUT and ENO is FALSE.	<table border="1"> <thead> <tr> <th>IN</th><th>OUT</th></tr> </thead> <tbody> <tr> <td>2.5</td><td>16#0003</td></tr> <tr> <td>1.4</td><td>16#0001</td></tr> <tr> <td>-2.5</td><td>16#FFFD</td></tr> <tr> <td>-1.4</td><td>16#FFFF</td></tr> <tr> <td>-8.4</td><td>16#FFF8</td></tr> </tbody> </table>	IN	OUT	2.5	16#0003	1.4	16#0001	-2.5	16#FFFD	-1.4	16#FFFF	-8.4	16#FFF8																																					
IN	OUT																																																	
2.5	16#0003																																																	
1.4	16#0001																																																	
-2.5	16#FFFD																																																	
-1.4	16#FFFF																																																	
-8.4	16#FFF8																																																	
If the modulus of the rounded value of IN is greater than the value that can be represented with WORD, but less than $2^{32} - 1$, IN is limited to [IN modulus 65536.0] and ENO is FALSE. If the modulus of the rounded value of IN is greater than or equal to 2^{32} FFFF is output and ENO is FALSE.	<table border="1"> <thead> <tr> <th>IN</th><th>OUT</th></tr> </thead> <tbody> <tr> <td>97680.0</td><td>16#7D90</td></tr> <tr> <td>1.1e+23</td><td>16#FFFF</td></tr> </tbody> </table>	IN	OUT	97680.0	16#7D90	1.1e+23	16#FFFF																																											
IN	OUT																																																	
97680.0	16#7D90																																																	
1.1e+23	16#FFFF																																																	

Output	Data type	Description													
OUT	WORD	Input data type: TIME <table border="1"> <thead> <tr> <th>Description</th><th>Examples</th></tr> </thead> <tbody> <tr> <td>If the IN data type is TIME, the value of IN is interpreted in milliseconds [ms] and converted to WORD. If IN is greater than the value that can be represented with WORD, IN is limited to [IN modulus 65536] and ENO is FALSE. If IN is negative, the corresponding two's complement is output to OUT and ENO is set to FALSE. Whenever the value of IN is greater than or equal to 2^{32} ms, FFFF is output and ENO is FALSE.</td><td> <table border="1"> <thead> <tr> <th>IN</th><th>OUT</th></tr> </thead> <tbody> <tr> <td>T#32144ms</td><td>16#7D90</td></tr> <tr> <td>T#97680ms</td><td>16#7D90</td></tr> <tr> <td>T#-1024ms</td><td>16#FC00</td></tr> </tbody> </table> </td></tr> </tbody> </table>		Description	Examples	If the IN data type is TIME, the value of IN is interpreted in milliseconds [ms] and converted to WORD. If IN is greater than the value that can be represented with WORD, IN is limited to [IN modulus 65536] and ENO is FALSE. If IN is negative, the corresponding two's complement is output to OUT and ENO is set to FALSE. Whenever the value of IN is greater than or equal to 2^{32} ms, FFFF is output and ENO is FALSE.	<table border="1"> <thead> <tr> <th>IN</th><th>OUT</th></tr> </thead> <tbody> <tr> <td>T#32144ms</td><td>16#7D90</td></tr> <tr> <td>T#97680ms</td><td>16#7D90</td></tr> <tr> <td>T#-1024ms</td><td>16#FC00</td></tr> </tbody> </table>	IN	OUT	T#32144ms	16#7D90	T#97680ms	16#7D90	T#-1024ms	16#FC00
Description	Examples														
If the IN data type is TIME, the value of IN is interpreted in milliseconds [ms] and converted to WORD. If IN is greater than the value that can be represented with WORD, IN is limited to [IN modulus 65536] and ENO is FALSE. If IN is negative, the corresponding two's complement is output to OUT and ENO is set to FALSE. Whenever the value of IN is greater than or equal to 2^{32} ms, FFFF is output and ENO is FALSE.	<table border="1"> <thead> <tr> <th>IN</th><th>OUT</th></tr> </thead> <tbody> <tr> <td>T#32144ms</td><td>16#7D90</td></tr> <tr> <td>T#97680ms</td><td>16#7D90</td></tr> <tr> <td>T#-1024ms</td><td>16#FC00</td></tr> </tbody> </table>	IN	OUT	T#32144ms	16#7D90	T#97680ms	16#7D90	T#-1024ms	16#FC00						
IN	OUT														
T#32144ms	16#7D90														
T#97680ms	16#7D90														
T#-1024ms	16#FC00														
<u>ENO</u>	BOOL	<table border="1"> <tbody> <tr> <td>TRUE</td><td>If EN = TRUE and the function was processed without errors.</td></tr> <tr> <td>FALSE</td><td> <ul style="list-style-type: none"> • If also EN = FALSE or • Error: The result is out of the range of values for the WORD data type. </td></tr> </tbody> </table>		TRUE	If EN = TRUE and the function was processed without errors.	FALSE	<ul style="list-style-type: none"> • If also EN = FALSE or • Error: The result is out of the range of values for the WORD data type. 								
TRUE	If EN = TRUE and the function was processed without errors.														
FALSE	<ul style="list-style-type: none"> • If also EN = FALSE or • Error: The result is out of the range of values for the WORD data type. 														

5.4.3 Conversion Within the Ranges of Values

Data type at IN	Input lower limit	Output lower limit	Input upper limit	Output upper limit
BOOL	0	16#0000	1	16#0001
BYTE	16#00	16#0000	16#FF	16#00FF
DINT	0	16#0000	65535	16#FFFF
DWORD	16#00	16#0000	16#FFFF	16#FFFF
INT	0	16#0000	32767	16#7FFF
LREAL	0.0	16#0000	65535.0	16#FFFF
REAL	0.0	16#0000	65535.0	16#FFFF
SINT	0	16#0000	127	16#007F
TIME	T#0ms	16#0000	T#1m5s535ms	16#FFFF
UDINT	0	16#0000	65535	16#FFFF
UINT	0	16#0000	65535	16#FFFF
USINT	0	16#0000	255	16#00FF
WORD	16#00	16#0000	16#FFFF	16#FFFF

5.4.4 Conversion Outside the Ranges of Values

Data type at IN	Input lower limit - 1	Output lower limit	Input upper limit + 1	Output upper limit
BOOL	-/-	-/-	-/-	-/-
BYTE	-/-	-/-	-/-	-/-
DINT	-1	16#FFFF	65536	16#0000
DWORD	-/-	-/-	16#010000	16#0000
INT	-1	16#FFFF	-/-	-/-
LREAL	-1.0	16#FFFF	65536.0	16#0000
REAL	-1.0	16#FFFF	65536.0	16#0000
SINT	-1	16#FFFF	-/-	-/-
TIME	T#-1ms	16#FFFF	T#65536ms	16#0000
UDINT	-/-	-/-	65536	16#0000
UINT	-/-	-/-	-/-	-/-
USINT	-/-	-/-	-/-	-/-
WORD	-/-	-/-	-/-	-/-

5.5 ATODWORD (Any to Double WORD)

5.5.1 Inputs

Input	Data type	Description				
IN	ANY	<p>Input IN</p>				
<u>EN</u>	BOOL	<p>EN is hidden by default. If EN is connected, it is always TRUE.</p> <p>Show EN/ENO: Right-click a function block or function, and select Show EN/ENO from the context menu.</p> <p>As soon as EN or ENO is connected, it can no longer be hidden.</p> <table border="1"> <tr> <td>TRUE</td><td> <p>The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle.</p> <p>ENO Output: The output is TRUE and can be set to FALSE separately based on the internal error detection.</p> </td></tr> <tr> <td>FALSE</td><td> <p>The function block or function is not processed by the user program. The following variables and value field assignment are not assigned.</p> <p>The outputs of a function block display the results of the user program cycle that was lastly processed.</p> <p>The outputs of a function display the default values defined for the data type.</p> <p>As long as the function block or function has not been completely processed, the outputs retain the default values defined for the data type.</p> <p>ENO Output: The output is also set to FALSE.</p> </td></tr> </table>	TRUE	<p>The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle.</p> <p>ENO Output: The output is TRUE and can be set to FALSE separately based on the internal error detection.</p>	FALSE	<p>The function block or function is not processed by the user program. The following variables and value field assignment are not assigned.</p> <p>The outputs of a function block display the results of the user program cycle that was lastly processed.</p> <p>The outputs of a function display the default values defined for the data type.</p> <p>As long as the function block or function has not been completely processed, the outputs retain the default values defined for the data type.</p> <p>ENO Output: The output is also set to FALSE.</p>
TRUE	<p>The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle.</p> <p>ENO Output: The output is TRUE and can be set to FALSE separately based on the internal error detection.</p>					
FALSE	<p>The function block or function is not processed by the user program. The following variables and value field assignment are not assigned.</p> <p>The outputs of a function block display the results of the user program cycle that was lastly processed.</p> <p>The outputs of a function display the default values defined for the data type.</p> <p>As long as the function block or function has not been completely processed, the outputs retain the default values defined for the data type.</p> <p>ENO Output: The output is also set to FALSE.</p>					

5.5.2 Outputs

Output	Data type	Description																																												
OUT	DWORD	<p>Converts the data type at IN to DWORD. The data type at IN is determined by creating a typecasting line. The following conditions apply to the value of OUT.</p> <p>Input data type: BOOL</p> <table border="1"> <thead> <tr> <th>Description</th><th>Examples</th></tr> </thead> <tbody> <tr> <td>The value TRUE and FALSE at IN are output to OUT as 1 and 0, respectively.</td><td> <table border="1"> <thead> <tr> <th>IN</th><th>OUT</th></tr> </thead> <tbody> <tr> <td>TRUE</td><td>1</td></tr> <tr> <td>FALSE</td><td>0</td></tr> </tbody> </table> </td></tr> </tbody> </table> <p>Input data type: ANY_INT</p> <table border="1"> <thead> <tr> <th>Description</th><th>Examples</th></tr> </thead> <tbody> <tr> <td>If the value at IN exceeds the range of values defined for DWORD, the portion that can be represented with the DWORD data type is output to OUT, and ENO is set to FALSE. If IN is negative, the corresponding two's complement is output to OUT.</td><td> <table border="1"> <thead> <tr> <th>IN</th><th>OUT</th></tr> </thead> <tbody> <tr> <td>16#FFFF</td><td>16#0000FFFF</td></tr> <tr> <td>$2^{32} + 1$</td><td>16#00000001</td></tr> <tr> <td>-3</td><td>16#FFFFFFFD</td></tr> </tbody> </table> </td></tr> </tbody> </table> <p>Input data type: ANY_REAL</p> <table border="1"> <thead> <tr> <th>Description</th><th>Examples</th></tr> </thead> <tbody> <tr> <td>In accordance with DIN 1333, REAL data types are rounded. If the first decimal place is equal to or greater than 5, the data type is rounded up, otherwise it is rounded down. If IN is negative, the corresponding two's complement is output to OUT.</td><td> <table border="1"> <thead> <tr> <th>IN</th><th>OUT</th></tr> </thead> <tbody> <tr> <td>2.5</td><td>16#00000003</td></tr> <tr> <td>1.4</td><td>16#00000001</td></tr> <tr> <td>-2.5</td><td>16#FFFFFFFD</td></tr> <tr> <td>-1.4</td><td>16#FFFFFFFF</td></tr> </tbody> </table> </td></tr> <tr> <td>Whenever the rounded value at IN is greater than $2^{32} - 1$ (16#FFFF FFFE), 16#FFFF FFFF is output. Whenever the rounded value of IN is less than $-2^{32} - 1$, 16#80000000 is output.</td><td> <table border="1"> <thead> <tr> <th>IN</th><th>OUT</th></tr> </thead> <tbody> <tr> <td>$5.5e+9$</td><td>16#FFFFFFFF</td></tr> <tr> <td>$-5.5e+9$</td><td>-16#80000000</td></tr> </tbody> </table> </td></tr> </tbody> </table>	Description	Examples	The value TRUE and FALSE at IN are output to OUT as 1 and 0, respectively.	<table border="1"> <thead> <tr> <th>IN</th><th>OUT</th></tr> </thead> <tbody> <tr> <td>TRUE</td><td>1</td></tr> <tr> <td>FALSE</td><td>0</td></tr> </tbody> </table>	IN	OUT	TRUE	1	FALSE	0	Description	Examples	If the value at IN exceeds the range of values defined for DWORD, the portion that can be represented with the DWORD data type is output to OUT, and ENO is set to FALSE. If IN is negative, the corresponding two's complement is output to OUT.	<table border="1"> <thead> <tr> <th>IN</th><th>OUT</th></tr> </thead> <tbody> <tr> <td>16#FFFF</td><td>16#0000FFFF</td></tr> <tr> <td>$2^{32} + 1$</td><td>16#00000001</td></tr> <tr> <td>-3</td><td>16#FFFFFFFD</td></tr> </tbody> </table>	IN	OUT	16#FFFF	16#0000FFFF	$2^{32} + 1$	16#00000001	-3	16#FFFFFFFD	Description	Examples	In accordance with DIN 1333, REAL data types are rounded. If the first decimal place is equal to or greater than 5, the data type is rounded up, otherwise it is rounded down. If IN is negative, the corresponding two's complement is output to OUT.	<table border="1"> <thead> <tr> <th>IN</th><th>OUT</th></tr> </thead> <tbody> <tr> <td>2.5</td><td>16#00000003</td></tr> <tr> <td>1.4</td><td>16#00000001</td></tr> <tr> <td>-2.5</td><td>16#FFFFFFFD</td></tr> <tr> <td>-1.4</td><td>16#FFFFFFFF</td></tr> </tbody> </table>	IN	OUT	2.5	16#00000003	1.4	16#00000001	-2.5	16#FFFFFFFD	-1.4	16#FFFFFFFF	Whenever the rounded value at IN is greater than $2^{32} - 1$ (16#FFFF FFFE), 16#FFFF FFFF is output. Whenever the rounded value of IN is less than $-2^{32} - 1$, 16#80000000 is output.	<table border="1"> <thead> <tr> <th>IN</th><th>OUT</th></tr> </thead> <tbody> <tr> <td>$5.5e+9$</td><td>16#FFFFFFFF</td></tr> <tr> <td>$-5.5e+9$</td><td>-16#80000000</td></tr> </tbody> </table>	IN	OUT	$5.5e+9$	16#FFFFFFFF	$-5.5e+9$	-16#80000000
Description	Examples																																													
The value TRUE and FALSE at IN are output to OUT as 1 and 0, respectively.	<table border="1"> <thead> <tr> <th>IN</th><th>OUT</th></tr> </thead> <tbody> <tr> <td>TRUE</td><td>1</td></tr> <tr> <td>FALSE</td><td>0</td></tr> </tbody> </table>	IN	OUT	TRUE	1	FALSE	0																																							
IN	OUT																																													
TRUE	1																																													
FALSE	0																																													
Description	Examples																																													
If the value at IN exceeds the range of values defined for DWORD, the portion that can be represented with the DWORD data type is output to OUT, and ENO is set to FALSE. If IN is negative, the corresponding two's complement is output to OUT.	<table border="1"> <thead> <tr> <th>IN</th><th>OUT</th></tr> </thead> <tbody> <tr> <td>16#FFFF</td><td>16#0000FFFF</td></tr> <tr> <td>$2^{32} + 1$</td><td>16#00000001</td></tr> <tr> <td>-3</td><td>16#FFFFFFFD</td></tr> </tbody> </table>	IN	OUT	16#FFFF	16#0000FFFF	$2^{32} + 1$	16#00000001	-3	16#FFFFFFFD																																					
IN	OUT																																													
16#FFFF	16#0000FFFF																																													
$2^{32} + 1$	16#00000001																																													
-3	16#FFFFFFFD																																													
Description	Examples																																													
In accordance with DIN 1333, REAL data types are rounded. If the first decimal place is equal to or greater than 5, the data type is rounded up, otherwise it is rounded down. If IN is negative, the corresponding two's complement is output to OUT.	<table border="1"> <thead> <tr> <th>IN</th><th>OUT</th></tr> </thead> <tbody> <tr> <td>2.5</td><td>16#00000003</td></tr> <tr> <td>1.4</td><td>16#00000001</td></tr> <tr> <td>-2.5</td><td>16#FFFFFFFD</td></tr> <tr> <td>-1.4</td><td>16#FFFFFFFF</td></tr> </tbody> </table>	IN	OUT	2.5	16#00000003	1.4	16#00000001	-2.5	16#FFFFFFFD	-1.4	16#FFFFFFFF																																			
IN	OUT																																													
2.5	16#00000003																																													
1.4	16#00000001																																													
-2.5	16#FFFFFFFD																																													
-1.4	16#FFFFFFFF																																													
Whenever the rounded value at IN is greater than $2^{32} - 1$ (16#FFFF FFFE), 16#FFFF FFFF is output. Whenever the rounded value of IN is less than $-2^{32} - 1$, 16#80000000 is output.	<table border="1"> <thead> <tr> <th>IN</th><th>OUT</th></tr> </thead> <tbody> <tr> <td>$5.5e+9$</td><td>16#FFFFFFFF</td></tr> <tr> <td>$-5.5e+9$</td><td>-16#80000000</td></tr> </tbody> </table>	IN	OUT	$5.5e+9$	16#FFFFFFFF	$-5.5e+9$	-16#80000000																																							
IN	OUT																																													
$5.5e+9$	16#FFFFFFFF																																													
$-5.5e+9$	-16#80000000																																													

Output	Data type	Description											
		Input data type: TIME											
OUT	DWORD	<table border="1"> <thead> <tr> <th>Description</th><th colspan="2">Examples</th></tr> </thead> <tbody> <tr> <td>If the IN data type is TIME, the value at IN is interpreted in milliseconds [ms] and converted to DINT.</td><td>IN</td><td>OUT</td></tr> <tr> <td></td><td>T#127ms</td><td>127</td></tr> </tbody> </table>			Description	Examples		If the IN data type is TIME, the value at IN is interpreted in milliseconds [ms] and converted to DINT.	IN	OUT		T#127ms	127
Description	Examples												
If the IN data type is TIME, the value at IN is interpreted in milliseconds [ms] and converted to DINT.	IN	OUT											
	T#127ms	127											
<u>ENO</u>	BOOL	TRUE	If EN = TRUE and the function was processed without errors.										
		FALSE	<ul style="list-style-type: none"> • If also EN = FALSE or • Error: The result is out of the range of values for the DWORD data type. 										

5.5.3 Conversion Within the Ranges of Values

Data type at IN	Input lower limit	Output lower limit	Input upper limit	Output upper limit
BOOL	0	16#00	1	16#00000000
BYTE	16#00	16#00	16#FF	16#000000FF
DINT	0	16#00	2147483647	16#7FFFFFFF
DWORD	16#00	16#00	16#FFFFFFFF	16#FFFFFFFF
INT	0	16#00	32767	16#00007FFF
LREAL	0.0	16#00	4294967295.0	16#FFFFFFFF
REAL	0.0	16#00	4.29496e+09	16#FFFFE380
SINT	0	16#00	127	16#0000007F
TIME	T#0ms	16#00	T#49d17h2m47s295ms	16#FFFFFFFF
UDINT	0	16#00	4294967295	16#FFFFFFFF
UINT	0	16#00	65535	16#0000FFFF
USINT	0	16#00	255	16#000000FF
WORD	16#00	16#00	16#FFFF	16#0000FFFF

5.5.4 Conversion Outside the Ranges of Values

Data type at IN	Input lower limit - 1	Output lower limit	Input upper limit + 1	Output upper limit
BOOL	-/-	-/-	-/-	-/-
BYTE	-/-	-/-	-/-	-/-
DINT	-1	16#FFFFFF	-/-	-/-
DWORD	-/-	-/-	-/-	-/-
INT	-1	16#FFFFFF	-/-	-/-
LREAL	-1.0	16#80000000	4294967296.0	16#FFFFFF
REAL	-1.0	16#80000000	4294967296.0	16#FFFFFF
SINT	-1	16#FFFFFF	-/-	-/-
TIME	T#-1ms	16#FFFFFF	T#49d17h2m47s296ms	16#00000000
UDINT	-/-	-/-	-/-	-/-
UINT	-/-	-/-	-/-	-/-
USINT	-/-	-/-	-/-	-/-
WORD	-/-	-/-	-/-	-/-

5.6 ATOLWORD (Any to Long WORD)

5.6.1 Inputs

Input	Data type	Description				
IN	ANY ELEMENTARY	<p>Input IN</p>				
<u>EN</u>	BOOL	<p>EN is hidden by default. If EN is connected, it is always TRUE.</p> <p>Show EN/ENO: Right-click a function block or function, and select Show EN/ENO from the context menu.</p> <p>As soon as EN or ENO is connected, it can no longer be hidden.</p> <table border="1"> <tr> <td>TRUE</td><td> <p>The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle.</p> <p>ENO Output: The output is TRUE and can be set to FALSE separately based on the internal error detection.</p> </td></tr> <tr> <td>FALSE</td><td> <p>The function block or function is not processed by the user program. The following variables and value field assignment are not assigned.</p> <p>The outputs of a function block display the results of the user program cycle that was lastly processed.</p> <p>The outputs of a function display the default values defined for the data type.</p> <p>As long as the function block or function has not been completely processed, the outputs retain the default values defined for the data type.</p> <p>ENO Output: The output is also set to FALSE.</p> </td></tr> </table>	TRUE	<p>The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle.</p> <p>ENO Output: The output is TRUE and can be set to FALSE separately based on the internal error detection.</p>	FALSE	<p>The function block or function is not processed by the user program. The following variables and value field assignment are not assigned.</p> <p>The outputs of a function block display the results of the user program cycle that was lastly processed.</p> <p>The outputs of a function display the default values defined for the data type.</p> <p>As long as the function block or function has not been completely processed, the outputs retain the default values defined for the data type.</p> <p>ENO Output: The output is also set to FALSE.</p>
TRUE	<p>The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle.</p> <p>ENO Output: The output is TRUE and can be set to FALSE separately based on the internal error detection.</p>					
FALSE	<p>The function block or function is not processed by the user program. The following variables and value field assignment are not assigned.</p> <p>The outputs of a function block display the results of the user program cycle that was lastly processed.</p> <p>The outputs of a function display the default values defined for the data type.</p> <p>As long as the function block or function has not been completely processed, the outputs retain the default values defined for the data type.</p> <p>ENO Output: The output is also set to FALSE.</p>					

5.6.2 Outputs

Output	Data type	Description																																																
OUT	LWORD	<p>Converts the data type at IN to LWORD. The data type at IN is determined by creating a typecasting line.</p> <p>Input data type: BOOL</p> <table border="1"> <thead> <tr> <th>Description</th><th>Examples</th></tr> </thead> <tbody> <tr> <td>The value TRUE and FALSE at IN are output to OUT as 1 and 0, respectively.</td><td> <table border="1"> <thead> <tr> <th>IN</th><th>OUT</th></tr> </thead> <tbody> <tr> <td>TRUE</td><td>16#0000000000000001</td></tr> <tr> <td>FALSE</td><td>16#0000000000000000</td></tr> </tbody> </table> </td></tr> </tbody> </table> <p>Input data type: ANY_INT</p> <table border="1"> <thead> <tr> <th>Description</th><th>Examples</th></tr> </thead> <tbody> <tr> <td>ULINT ($2^{64} - 1$) is still within the range of values for LWORD. If IN is negative, the corresponding two's complement is output to OUT and ENO is FALSE.</td><td> <table border="1"> <thead> <tr> <th>IN</th><th>OUT</th></tr> </thead> <tbody> <tr> <td>1024</td><td>16#0000000000000400</td></tr> <tr> <td>$2^{64} - 1$</td><td>16#FFFFFFFFFFFFFFF</td></tr> <tr> <td>-1</td><td>16#FFFFFFFFFFFFFFF</td></tr> <tr> <td>-2</td><td>16#FFFFFFFFFFFFFFF</td></tr> </tbody> </table> </td></tr> </tbody> </table> <p>Input data type: ANY_REAL</p> <table border="1"> <thead> <tr> <th>Description</th><th>Examples</th></tr> </thead> <tbody> <tr> <td>In accordance with DIN 1333, REAL data types are rounded. If the first decimal place is equal to or greater than 5, the data type is rounded up, otherwise it is rounded down. If IN is negative, the corresponding two's complement is output to OUT and ENO is FALSE.</td><td> <table border="1"> <thead> <tr> <th>IN</th><th>OUT</th></tr> </thead> <tbody> <tr> <td>2.5</td><td>16#0000000000000003</td></tr> <tr> <td>1.4</td><td>16#0000000000000001</td></tr> <tr> <td>-2.5</td><td>16#FFFFFFFFFFFFFFFD</td></tr> <tr> <td>-1.4</td><td>16#FFFFFFFFFFFFFFF</td></tr> <tr> <td>-8.4</td><td>16#FFFFFFFFFFFFFFF8</td></tr> </tbody> </table> </td></tr> <tr> <td>If the modulus of the rounded value at IN is greater than the value that can be represented with LWORD, OUT contains the largest value that can be represented with the LWORD data type, and ENO is FALSE.</td><td> <table border="1"> <thead> <tr> <th>IN</th><th>OUT</th></tr> </thead> <tbody> <tr> <td>$1.9e+1 - 9$</td><td>16#FFFFFFFFFFFFFF-F</td></tr> <tr> <td>$1.1e+2 - 3$</td><td>16#FFFFFFFFFFFFFF-F</td></tr> </tbody> </table> </td></tr> </tbody> </table>	Description	Examples	The value TRUE and FALSE at IN are output to OUT as 1 and 0, respectively.	<table border="1"> <thead> <tr> <th>IN</th><th>OUT</th></tr> </thead> <tbody> <tr> <td>TRUE</td><td>16#0000000000000001</td></tr> <tr> <td>FALSE</td><td>16#0000000000000000</td></tr> </tbody> </table>	IN	OUT	TRUE	16#0000000000000001	FALSE	16#0000000000000000	Description	Examples	ULINT ($2^{64} - 1$) is still within the range of values for LWORD. If IN is negative, the corresponding two's complement is output to OUT and ENO is FALSE.	<table border="1"> <thead> <tr> <th>IN</th><th>OUT</th></tr> </thead> <tbody> <tr> <td>1024</td><td>16#0000000000000400</td></tr> <tr> <td>$2^{64} - 1$</td><td>16#FFFFFFFFFFFFFFF</td></tr> <tr> <td>-1</td><td>16#FFFFFFFFFFFFFFF</td></tr> <tr> <td>-2</td><td>16#FFFFFFFFFFFFFFF</td></tr> </tbody> </table>	IN	OUT	1024	16#0000000000000400	$2^{64} - 1$	16#FFFFFFFFFFFFFFF	-1	16#FFFFFFFFFFFFFFF	-2	16#FFFFFFFFFFFFFFF	Description	Examples	In accordance with DIN 1333, REAL data types are rounded. If the first decimal place is equal to or greater than 5, the data type is rounded up, otherwise it is rounded down. If IN is negative, the corresponding two's complement is output to OUT and ENO is FALSE.	<table border="1"> <thead> <tr> <th>IN</th><th>OUT</th></tr> </thead> <tbody> <tr> <td>2.5</td><td>16#0000000000000003</td></tr> <tr> <td>1.4</td><td>16#0000000000000001</td></tr> <tr> <td>-2.5</td><td>16#FFFFFFFFFFFFFFFD</td></tr> <tr> <td>-1.4</td><td>16#FFFFFFFFFFFFFFF</td></tr> <tr> <td>-8.4</td><td>16#FFFFFFFFFFFFFFF8</td></tr> </tbody> </table>	IN	OUT	2.5	16#0000000000000003	1.4	16#0000000000000001	-2.5	16#FFFFFFFFFFFFFFFD	-1.4	16#FFFFFFFFFFFFFFF	-8.4	16#FFFFFFFFFFFFFFF8	If the modulus of the rounded value at IN is greater than the value that can be represented with LWORD, OUT contains the largest value that can be represented with the LWORD data type, and ENO is FALSE.	<table border="1"> <thead> <tr> <th>IN</th><th>OUT</th></tr> </thead> <tbody> <tr> <td>$1.9e+1 - 9$</td><td>16#FFFFFFFFFFFFFF-F</td></tr> <tr> <td>$1.1e+2 - 3$</td><td>16#FFFFFFFFFFFFFF-F</td></tr> </tbody> </table>	IN	OUT	$1.9e+1 - 9$	16#FFFFFFFFFFFFFF-F	$1.1e+2 - 3$	16#FFFFFFFFFFFFFF-F
Description	Examples																																																	
The value TRUE and FALSE at IN are output to OUT as 1 and 0, respectively.	<table border="1"> <thead> <tr> <th>IN</th><th>OUT</th></tr> </thead> <tbody> <tr> <td>TRUE</td><td>16#0000000000000001</td></tr> <tr> <td>FALSE</td><td>16#0000000000000000</td></tr> </tbody> </table>	IN	OUT	TRUE	16#0000000000000001	FALSE	16#0000000000000000																																											
IN	OUT																																																	
TRUE	16#0000000000000001																																																	
FALSE	16#0000000000000000																																																	
Description	Examples																																																	
ULINT ($2^{64} - 1$) is still within the range of values for LWORD. If IN is negative, the corresponding two's complement is output to OUT and ENO is FALSE.	<table border="1"> <thead> <tr> <th>IN</th><th>OUT</th></tr> </thead> <tbody> <tr> <td>1024</td><td>16#0000000000000400</td></tr> <tr> <td>$2^{64} - 1$</td><td>16#FFFFFFFFFFFFFFF</td></tr> <tr> <td>-1</td><td>16#FFFFFFFFFFFFFFF</td></tr> <tr> <td>-2</td><td>16#FFFFFFFFFFFFFFF</td></tr> </tbody> </table>	IN	OUT	1024	16#0000000000000400	$2^{64} - 1$	16#FFFFFFFFFFFFFFF	-1	16#FFFFFFFFFFFFFFF	-2	16#FFFFFFFFFFFFFFF																																							
IN	OUT																																																	
1024	16#0000000000000400																																																	
$2^{64} - 1$	16#FFFFFFFFFFFFFFF																																																	
-1	16#FFFFFFFFFFFFFFF																																																	
-2	16#FFFFFFFFFFFFFFF																																																	
Description	Examples																																																	
In accordance with DIN 1333, REAL data types are rounded. If the first decimal place is equal to or greater than 5, the data type is rounded up, otherwise it is rounded down. If IN is negative, the corresponding two's complement is output to OUT and ENO is FALSE.	<table border="1"> <thead> <tr> <th>IN</th><th>OUT</th></tr> </thead> <tbody> <tr> <td>2.5</td><td>16#0000000000000003</td></tr> <tr> <td>1.4</td><td>16#0000000000000001</td></tr> <tr> <td>-2.5</td><td>16#FFFFFFFFFFFFFFFD</td></tr> <tr> <td>-1.4</td><td>16#FFFFFFFFFFFFFFF</td></tr> <tr> <td>-8.4</td><td>16#FFFFFFFFFFFFFFF8</td></tr> </tbody> </table>	IN	OUT	2.5	16#0000000000000003	1.4	16#0000000000000001	-2.5	16#FFFFFFFFFFFFFFFD	-1.4	16#FFFFFFFFFFFFFFF	-8.4	16#FFFFFFFFFFFFFFF8																																					
IN	OUT																																																	
2.5	16#0000000000000003																																																	
1.4	16#0000000000000001																																																	
-2.5	16#FFFFFFFFFFFFFFFD																																																	
-1.4	16#FFFFFFFFFFFFFFF																																																	
-8.4	16#FFFFFFFFFFFFFFF8																																																	
If the modulus of the rounded value at IN is greater than the value that can be represented with LWORD, OUT contains the largest value that can be represented with the LWORD data type, and ENO is FALSE.	<table border="1"> <thead> <tr> <th>IN</th><th>OUT</th></tr> </thead> <tbody> <tr> <td>$1.9e+1 - 9$</td><td>16#FFFFFFFFFFFFFF-F</td></tr> <tr> <td>$1.1e+2 - 3$</td><td>16#FFFFFFFFFFFFFF-F</td></tr> </tbody> </table>	IN	OUT	$1.9e+1 - 9$	16#FFFFFFFFFFFFFF-F	$1.1e+2 - 3$	16#FFFFFFFFFFFFFF-F																																											
IN	OUT																																																	
$1.9e+1 - 9$	16#FFFFFFFFFFFFFF-F																																																	
$1.1e+2 - 3$	16#FFFFFFFFFFFFFF-F																																																	

Output	Data type	Description																		
OUT	LWORD	Input data type: TIME <table border="1"> <thead> <tr> <th colspan="2">Description</th> <th colspan="2">Examples</th> </tr> </thead> <tbody> <tr> <td colspan="2"> If the IN data type is TIME, the value of IN is interpreted in milliseconds [ms] and converted to LWORD. If IN is negative, the corresponding two's complement is output to OUT and ENO is FALSE. </td> <td>IN</td> <td>OUT</td> </tr> <tr> <td colspan="2"></td> <td>T#1024ms</td> <td>16#00000000▼ 00000400</td> </tr> <tr> <td colspan="2"></td> <td>T#-1024ms</td> <td>16#FFFFFFFF▼ FFFFFC00</td> </tr> </tbody> </table>			Description		Examples		If the IN data type is TIME, the value of IN is interpreted in milliseconds [ms] and converted to LWORD. If IN is negative, the corresponding two's complement is output to OUT and ENO is FALSE.		IN	OUT			T#1024ms	16#00000000▼ 00000400			T#-1024ms	16#FFFFFFFF▼ FFFFFC00
Description		Examples																		
If the IN data type is TIME, the value of IN is interpreted in milliseconds [ms] and converted to LWORD. If IN is negative, the corresponding two's complement is output to OUT and ENO is FALSE.		IN	OUT																	
		T#1024ms	16#00000000▼ 00000400																	
		T#-1024ms	16#FFFFFFFF▼ FFFFFC00																	
<u>ENO</u>	BOOL	<table border="1"> <tr> <td>TRUE</td> <td colspan="2">If EN = TRUE and the function was processed without errors.</td> </tr> <tr> <td>FALSE</td> <td colspan="2"> <ul style="list-style-type: none"> If also EN = FALSE or Error: The result is out of the range of values for the LWORD data type. </td> </tr> </table>			TRUE	If EN = TRUE and the function was processed without errors.		FALSE	<ul style="list-style-type: none"> If also EN = FALSE or Error: The result is out of the range of values for the LWORD data type. 											
TRUE	If EN = TRUE and the function was processed without errors.																			
FALSE	<ul style="list-style-type: none"> If also EN = FALSE or Error: The result is out of the range of values for the LWORD data type. 																			

5.7 ATOUSINT (Any to Unsigned Single INT)

5.7.1 Inputs

Input	Data type	Description					
IN	ANY ELEMENTARY	Input IN					
<u>EN</u>	BOOL	<p>EN is hidden by default. If EN is connected, it is always TRUE.</p> <p>Show EN/ENO: Right-click a function block or function, and select Show EN/ENO from the context menu.</p> <p>As soon as EN or ENO is connected, it can no longer be hidden.</p> <table border="1"> <tr> <td>TRUE</td> <td> The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle. ENO Output: The output is TRUE and can be set to FALSE separately based on the internal error detection. </td> </tr> <tr> <td>FALSE</td> <td> The function block or function is not processed by the user program. The following variables and value field assignment are not assigned. The outputs of a function block display the results of the user program cycle that was lastly processed. The outputs of a function display the default values defined for the data type. As long as the function block or function has not been completely processed, the outputs retain the default values defined for the data type. ENO Output: The output is also set to FALSE. </td> </tr> </table>		TRUE	The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle. ENO Output: The output is TRUE and can be set to FALSE separately based on the internal error detection.	FALSE	The function block or function is not processed by the user program. The following variables and value field assignment are not assigned. The outputs of a function block display the results of the user program cycle that was lastly processed. The outputs of a function display the default values defined for the data type. As long as the function block or function has not been completely processed, the outputs retain the default values defined for the data type. ENO Output: The output is also set to FALSE.
TRUE	The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle. ENO Output: The output is TRUE and can be set to FALSE separately based on the internal error detection.						
FALSE	The function block or function is not processed by the user program. The following variables and value field assignment are not assigned. The outputs of a function block display the results of the user program cycle that was lastly processed. The outputs of a function display the default values defined for the data type. As long as the function block or function has not been completely processed, the outputs retain the default values defined for the data type. ENO Output: The output is also set to FALSE.						

5.7.2 Outputs

Output	Data type	Description																						
OUT	USINT	<p>Converts the data type of IN to USINT. The data type at IN is determined by creating a typecasting line.</p> <p>Input data type: BOOL</p> <table border="1"> <thead> <tr> <th>Description</th><th>Examples</th></tr> </thead> <tbody> <tr> <td>The value TRUE and FALSE at IN are output to OUT as 1 and 0, respectively.</td><td> <table border="1"> <thead> <tr> <th>IN</th><th>OUT</th></tr> </thead> <tbody> <tr> <td>TRUE</td><td>1</td></tr> <tr> <td>FALSE</td><td>0</td></tr> </tbody> </table> </td></tr> </tbody> </table>	Description	Examples	The value TRUE and FALSE at IN are output to OUT as 1 and 0, respectively.	<table border="1"> <thead> <tr> <th>IN</th><th>OUT</th></tr> </thead> <tbody> <tr> <td>TRUE</td><td>1</td></tr> <tr> <td>FALSE</td><td>0</td></tr> </tbody> </table>	IN	OUT	TRUE	1	FALSE	0												
Description	Examples																							
The value TRUE and FALSE at IN are output to OUT as 1 and 0, respectively.	<table border="1"> <thead> <tr> <th>IN</th><th>OUT</th></tr> </thead> <tbody> <tr> <td>TRUE</td><td>1</td></tr> <tr> <td>FALSE</td><td>0</td></tr> </tbody> </table>	IN	OUT	TRUE	1	FALSE	0																	
IN	OUT																							
TRUE	1																							
FALSE	0																							
Input data type: ANY_INT																								
Input data type: ANY_REAL																								
<table border="1"> <thead> <tr> <th>Description</th><th>Examples</th></tr> </thead> <tbody> <tr> <td> <p>If IN is greater than the value that can be represented with USINT, IN is limited to [IN modulus 256] and ENO is FALSE.</p> <p>If IN is negative, the corresponding two's complement is output to OUT.</p> </td><td> <table border="1"> <thead> <tr> <th>IN</th><th>OUT</th></tr> </thead> <tbody> <tr> <td>256</td><td>0</td></tr> <tr> <td>$2^{64} - 1$</td><td>255</td></tr> <tr> <td>-129</td><td>127</td></tr> <tr> <td>-2^{32}</td><td>0</td></tr> </tbody> </table> </td></tr> </tbody> </table>			Description	Examples	<p>If IN is greater than the value that can be represented with USINT, IN is limited to [IN modulus 256] and ENO is FALSE.</p> <p>If IN is negative, the corresponding two's complement is output to OUT.</p>	<table border="1"> <thead> <tr> <th>IN</th><th>OUT</th></tr> </thead> <tbody> <tr> <td>256</td><td>0</td></tr> <tr> <td>$2^{64} - 1$</td><td>255</td></tr> <tr> <td>-129</td><td>127</td></tr> <tr> <td>-2^{32}</td><td>0</td></tr> </tbody> </table>	IN	OUT	256	0	$2^{64} - 1$	255	-129	127	-2^{32}	0								
Description	Examples																							
<p>If IN is greater than the value that can be represented with USINT, IN is limited to [IN modulus 256] and ENO is FALSE.</p> <p>If IN is negative, the corresponding two's complement is output to OUT.</p>	<table border="1"> <thead> <tr> <th>IN</th><th>OUT</th></tr> </thead> <tbody> <tr> <td>256</td><td>0</td></tr> <tr> <td>$2^{64} - 1$</td><td>255</td></tr> <tr> <td>-129</td><td>127</td></tr> <tr> <td>-2^{32}</td><td>0</td></tr> </tbody> </table>	IN	OUT	256	0	$2^{64} - 1$	255	-129	127	-2^{32}	0													
IN	OUT																							
256	0																							
$2^{64} - 1$	255																							
-129	127																							
-2^{32}	0																							
<table border="1"> <thead> <tr> <th>Description</th><th>Examples</th></tr> </thead> <tbody> <tr> <td> <p>In accordance with DIN 1333, REAL data types are rounded. If the first decimal place is equal to or greater than 5, the data type is rounded up, otherwise it is rounded down.</p> <p>Whenever IN is negative, 0 is output.</p> </td><td> <table border="1"> <thead> <tr> <th>IN</th><th>OUT</th></tr> </thead> <tbody> <tr> <td>2.5</td><td>3</td></tr> <tr> <td>1.4</td><td>1</td></tr> <tr> <td>-2.5</td><td>0</td></tr> <tr> <td>-1.4</td><td>0</td></tr> </tbody> </table> </td></tr> <tr> <td> <p>If rounded value of IN is greater than the value that can be represented with USINT but less than $2^{32} - 1$, IN is limited to [IN modulus 256] and ENO is FALSE.</p> <p>If the rounded value of IN is greater than $2^{32} - 1$ (16#FFFF FFFF), 255 is output and ENO is FALSE.</p> </td><td> <table border="1"> <thead> <tr> <th>IN</th><th>OUT</th></tr> </thead> <tbody> <tr> <td>32760.0</td><td>248</td></tr> <tr> <td>$2^{32} + 5$</td><td>255</td></tr> </tbody> </table> </td></tr> </tbody> </table>			Description	Examples	<p>In accordance with DIN 1333, REAL data types are rounded. If the first decimal place is equal to or greater than 5, the data type is rounded up, otherwise it is rounded down.</p> <p>Whenever IN is negative, 0 is output.</p>	<table border="1"> <thead> <tr> <th>IN</th><th>OUT</th></tr> </thead> <tbody> <tr> <td>2.5</td><td>3</td></tr> <tr> <td>1.4</td><td>1</td></tr> <tr> <td>-2.5</td><td>0</td></tr> <tr> <td>-1.4</td><td>0</td></tr> </tbody> </table>	IN	OUT	2.5	3	1.4	1	-2.5	0	-1.4	0	<p>If rounded value of IN is greater than the value that can be represented with USINT but less than $2^{32} - 1$, IN is limited to [IN modulus 256] and ENO is FALSE.</p> <p>If the rounded value of IN is greater than $2^{32} - 1$ (16#FFFF FFFF), 255 is output and ENO is FALSE.</p>	<table border="1"> <thead> <tr> <th>IN</th><th>OUT</th></tr> </thead> <tbody> <tr> <td>32760.0</td><td>248</td></tr> <tr> <td>$2^{32} + 5$</td><td>255</td></tr> </tbody> </table>	IN	OUT	32760.0	248	$2^{32} + 5$	255
Description	Examples																							
<p>In accordance with DIN 1333, REAL data types are rounded. If the first decimal place is equal to or greater than 5, the data type is rounded up, otherwise it is rounded down.</p> <p>Whenever IN is negative, 0 is output.</p>	<table border="1"> <thead> <tr> <th>IN</th><th>OUT</th></tr> </thead> <tbody> <tr> <td>2.5</td><td>3</td></tr> <tr> <td>1.4</td><td>1</td></tr> <tr> <td>-2.5</td><td>0</td></tr> <tr> <td>-1.4</td><td>0</td></tr> </tbody> </table>	IN	OUT	2.5	3	1.4	1	-2.5	0	-1.4	0													
IN	OUT																							
2.5	3																							
1.4	1																							
-2.5	0																							
-1.4	0																							
<p>If rounded value of IN is greater than the value that can be represented with USINT but less than $2^{32} - 1$, IN is limited to [IN modulus 256] and ENO is FALSE.</p> <p>If the rounded value of IN is greater than $2^{32} - 1$ (16#FFFF FFFF), 255 is output and ENO is FALSE.</p>	<table border="1"> <thead> <tr> <th>IN</th><th>OUT</th></tr> </thead> <tbody> <tr> <td>32760.0</td><td>248</td></tr> <tr> <td>$2^{32} + 5$</td><td>255</td></tr> </tbody> </table>	IN	OUT	32760.0	248	$2^{32} + 5$	255																	
IN	OUT																							
32760.0	248																							
$2^{32} + 5$	255																							

Output	Data type	Description																	
OUT	USINT	<p>Input data type: TIME</p> <table border="1"> <thead> <tr> <th>Description</th><th>Examples</th></tr> </thead> <tbody> <tr> <td>IN must be within -2^{63} ms and $(2^{63} - 1)$ ms. If IN is less than 256, the value is interpreted in milliseconds [ms] and converted to USINT. If the value of IN is greater than the maximum value that can be represented with USINT, d, h, m and s are ignored. The ms portion of IN is limited to [IN modulus 256] and ENO is FALSE. If IN is negative, the two's complement is output to OUT and ENO is FALSE.</td><td> <table border="1"> <thead> <tr> <th>IN</th><th>OUT</th></tr> </thead> <tbody> <tr> <td>T#200ms</td><td>200</td></tr> <tr> <td>T#9d2h5m▼ 59s200ms</td><td>200</td></tr> <tr> <td>T#325ms</td><td>69</td></tr> <tr> <td>T#-325ms</td><td>187</td></tr> <tr> <td>T#-9d325ms</td><td>187</td></tr> </tbody> </table> </td></tr> </tbody> </table>		Description	Examples	IN must be within -2^{63} ms and $(2^{63} - 1)$ ms. If IN is less than 256, the value is interpreted in milliseconds [ms] and converted to USINT. If the value of IN is greater than the maximum value that can be represented with USINT, d, h, m and s are ignored. The ms portion of IN is limited to [IN modulus 256] and ENO is FALSE. If IN is negative, the two's complement is output to OUT and ENO is FALSE.	<table border="1"> <thead> <tr> <th>IN</th><th>OUT</th></tr> </thead> <tbody> <tr> <td>T#200ms</td><td>200</td></tr> <tr> <td>T#9d2h5m▼ 59s200ms</td><td>200</td></tr> <tr> <td>T#325ms</td><td>69</td></tr> <tr> <td>T#-325ms</td><td>187</td></tr> <tr> <td>T#-9d325ms</td><td>187</td></tr> </tbody> </table>	IN	OUT	T#200ms	200	T#9d2h5m▼ 59s200ms	200	T#325ms	69	T#-325ms	187	T#-9d325ms	187
Description	Examples																		
IN must be within -2^{63} ms and $(2^{63} - 1)$ ms. If IN is less than 256, the value is interpreted in milliseconds [ms] and converted to USINT. If the value of IN is greater than the maximum value that can be represented with USINT, d, h, m and s are ignored. The ms portion of IN is limited to [IN modulus 256] and ENO is FALSE. If IN is negative, the two's complement is output to OUT and ENO is FALSE.	<table border="1"> <thead> <tr> <th>IN</th><th>OUT</th></tr> </thead> <tbody> <tr> <td>T#200ms</td><td>200</td></tr> <tr> <td>T#9d2h5m▼ 59s200ms</td><td>200</td></tr> <tr> <td>T#325ms</td><td>69</td></tr> <tr> <td>T#-325ms</td><td>187</td></tr> <tr> <td>T#-9d325ms</td><td>187</td></tr> </tbody> </table>	IN	OUT	T#200ms	200	T#9d2h5m▼ 59s200ms	200	T#325ms	69	T#-325ms	187	T#-9d325ms	187						
IN	OUT																		
T#200ms	200																		
T#9d2h5m▼ 59s200ms	200																		
T#325ms	69																		
T#-325ms	187																		
T#-9d325ms	187																		
<u>ENO</u>	BOOL	<table border="1"> <tr> <td>TRUE</td><td>If EN = TRUE and the function was processed without errors.</td></tr> <tr> <td>FALSE</td><td> <ul style="list-style-type: none"> • If also EN = FALSE or • Error: The result is out of the range of values for USINT. </td></tr> </table>	TRUE	If EN = TRUE and the function was processed without errors.	FALSE	<ul style="list-style-type: none"> • If also EN = FALSE or • Error: The result is out of the range of values for USINT. 													
TRUE	If EN = TRUE and the function was processed without errors.																		
FALSE	<ul style="list-style-type: none"> • If also EN = FALSE or • Error: The result is out of the range of values for USINT. 																		

5.7.3 Conversion Within the Ranges of Values

Data type at IN	Input lower limit	Output lower limit	Input upper limit	Output upper limit
BOOL	0	0	1	1
BYTE	16#00	0	16#FF	255
DINT	0	0	255	255
DWORD	16#00	0	16#FF	255
INT	0	0	255	255
LREAL	0.0	0	255.0	255
REAL	0.0	0	255.0	255
SINT	0	0	127	127
TIME	T#0ms	0	T#255ms	255
UDINT	0	0	255	255
UINT	0	0	255	255
USINT	0	0	255	255
WORD	16#00	0	16#FF	255

5.7.4 Conversion Outside the Ranges of Values

Data type at IN	Input lower limit - 1	Output lower limit	Input upper limit + 1	Output upper limit
BOOL	-/-	-/-	-/-	-/-
BYTE	-/-	-/-	-/-	-/-
DINT	-1	255	256	0
DWORD	-/-	-/-	16#0100	0
INT	-1	255	256	0
LREAL	-1.0	255	256.0	0
REAL	-1.0	255	256.0	0
SINT	-1	255	-/-	-/-
TIME	T#-1ms	255	T#256ms	0
UDINT	-/-	-/-	256	0
UINT	-/-	-/-	256	0
USINT	-/-	-/-	-/-	-/-
WORD	-/-	-/-	16#0100	0

5.8 ATOSINT (Any to Single INT)

5.8.1 Inputs

Input	Data type	Description				
IN	ANY ELEMENTARY	<p>Input IN</p>				
EN	BOOL	<p>EN is hidden by default. If EN is connected, it is always TRUE.</p> <p>Show EN/ENO: Right-click a function block or function, and select Show EN/ENO from the context menu.</p> <p>As soon as EN or ENO is connected, it can no longer be hidden.</p> <table border="1"> <tr> <td>TRUE</td><td> <p>The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle.</p> <p>ENO Output: The output is TRUE and can be set to FALSE separately based on the internal error detection.</p> </td></tr> <tr> <td>FALSE</td><td> <p>The function block or function is not processed by the user program. The following variables and value field assignment are not assigned.</p> <p>The outputs of a function block display the results of the user program cycle that was lastly processed.</p> <p>The outputs of a function display the default values defined for the data type.</p> <p>As long as the function block or function has not been completely processed, the outputs retain the default values defined for the data type.</p> <p>ENO Output: The output is also set to FALSE.</p> </td></tr> </table>	TRUE	<p>The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle.</p> <p>ENO Output: The output is TRUE and can be set to FALSE separately based on the internal error detection.</p>	FALSE	<p>The function block or function is not processed by the user program. The following variables and value field assignment are not assigned.</p> <p>The outputs of a function block display the results of the user program cycle that was lastly processed.</p> <p>The outputs of a function display the default values defined for the data type.</p> <p>As long as the function block or function has not been completely processed, the outputs retain the default values defined for the data type.</p> <p>ENO Output: The output is also set to FALSE.</p>
TRUE	<p>The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle.</p> <p>ENO Output: The output is TRUE and can be set to FALSE separately based on the internal error detection.</p>					
FALSE	<p>The function block or function is not processed by the user program. The following variables and value field assignment are not assigned.</p> <p>The outputs of a function block display the results of the user program cycle that was lastly processed.</p> <p>The outputs of a function display the default values defined for the data type.</p> <p>As long as the function block or function has not been completely processed, the outputs retain the default values defined for the data type.</p> <p>ENO Output: The output is also set to FALSE.</p>					

5.8.2 Outputs

Output	Data type	Description																								
OUT	SINT	<p>Converts the data type at IN to SINT. The data type at IN is determined by creating a typecasting line.</p> <p>The following conditions apply to the value of OUT.</p> <p>Input data type: BOOL</p> <table border="1"> <thead> <tr> <th>Description</th><th>Examples</th></tr> </thead> <tbody> <tr> <td>The value TRUE and FALSE at IN are output to OUT as 1 and 0, respectively.</td><td> <table border="1"> <thead> <tr> <th>IN</th><th>OUT</th></tr> </thead> <tbody> <tr> <td>TRUE</td><td>1</td></tr> <tr> <td>FALSE</td><td>0</td></tr> </tbody> </table> </td></tr> </tbody> </table> <p>Input data type: ANY_INT</p> <table border="1"> <thead> <tr> <th>Description</th><th>Examples</th></tr> </thead> <tbody> <tr> <td>If the value of IN exceeds the range of values defined for SINT, the portion that can be represented with the SINT data type is output to OUT, and ENO is set to FALSE. If IN is positive and exceeds the upper range limit of SINT, OUT contains the corresponding two's complement. If IN is negative and exceeds the lower range limit of SINT, the corresponding two's complement is output to OUT.</td><td> <table border="1"> <thead> <tr> <th>IN</th><th>OUT</th></tr> </thead> <tbody> <tr> <td>129 (16#81)</td><td>-127</td></tr> <tr> <td>16#FF</td><td>-1</td></tr> <tr> <td>16#127F</td><td>7F</td></tr> <tr> <td>-129</td><td>127</td></tr> </tbody> </table> </td></tr> </tbody> </table>	Description	Examples	The value TRUE and FALSE at IN are output to OUT as 1 and 0, respectively.	<table border="1"> <thead> <tr> <th>IN</th><th>OUT</th></tr> </thead> <tbody> <tr> <td>TRUE</td><td>1</td></tr> <tr> <td>FALSE</td><td>0</td></tr> </tbody> </table>	IN	OUT	TRUE	1	FALSE	0	Description	Examples	If the value of IN exceeds the range of values defined for SINT, the portion that can be represented with the SINT data type is output to OUT, and ENO is set to FALSE. If IN is positive and exceeds the upper range limit of SINT, OUT contains the corresponding two's complement. If IN is negative and exceeds the lower range limit of SINT, the corresponding two's complement is output to OUT.	<table border="1"> <thead> <tr> <th>IN</th><th>OUT</th></tr> </thead> <tbody> <tr> <td>129 (16#81)</td><td>-127</td></tr> <tr> <td>16#FF</td><td>-1</td></tr> <tr> <td>16#127F</td><td>7F</td></tr> <tr> <td>-129</td><td>127</td></tr> </tbody> </table>	IN	OUT	129 (16#81)	-127	16#FF	-1	16#127F	7F	-129	127
Description	Examples																									
The value TRUE and FALSE at IN are output to OUT as 1 and 0, respectively.	<table border="1"> <thead> <tr> <th>IN</th><th>OUT</th></tr> </thead> <tbody> <tr> <td>TRUE</td><td>1</td></tr> <tr> <td>FALSE</td><td>0</td></tr> </tbody> </table>	IN	OUT	TRUE	1	FALSE	0																			
IN	OUT																									
TRUE	1																									
FALSE	0																									
Description	Examples																									
If the value of IN exceeds the range of values defined for SINT, the portion that can be represented with the SINT data type is output to OUT, and ENO is set to FALSE. If IN is positive and exceeds the upper range limit of SINT, OUT contains the corresponding two's complement. If IN is negative and exceeds the lower range limit of SINT, the corresponding two's complement is output to OUT.	<table border="1"> <thead> <tr> <th>IN</th><th>OUT</th></tr> </thead> <tbody> <tr> <td>129 (16#81)</td><td>-127</td></tr> <tr> <td>16#FF</td><td>-1</td></tr> <tr> <td>16#127F</td><td>7F</td></tr> <tr> <td>-129</td><td>127</td></tr> </tbody> </table>	IN	OUT	129 (16#81)	-127	16#FF	-1	16#127F	7F	-129	127															
IN	OUT																									
129 (16#81)	-127																									
16#FF	-1																									
16#127F	7F																									
-129	127																									

Output	Data type	Description																
		Input data type: ANY_REAL																
		<table border="1"> <thead> <tr> <th>Description</th><th>Examples</th></tr> </thead> <tbody> <tr> <td>In accordance with DIN 1333, REAL data types are rounded. If the first decimal place is equal to or greater than 5, the data type is rounded up, otherwise it is rounded down.</td><td> <table border="1"> <thead> <tr> <th>IN</th><th>OUT</th></tr> </thead> <tbody> <tr> <td>2.5</td><td>3</td></tr> <tr> <td>1.4</td><td>1</td></tr> <tr> <td>-2.5</td><td>-3</td></tr> <tr> <td>-1.4</td><td>-1</td></tr> </tbody> </table> </td></tr> </tbody> </table>			Description	Examples	In accordance with DIN 1333, REAL data types are rounded. If the first decimal place is equal to or greater than 5, the data type is rounded up, otherwise it is rounded down.	<table border="1"> <thead> <tr> <th>IN</th><th>OUT</th></tr> </thead> <tbody> <tr> <td>2.5</td><td>3</td></tr> <tr> <td>1.4</td><td>1</td></tr> <tr> <td>-2.5</td><td>-3</td></tr> <tr> <td>-1.4</td><td>-1</td></tr> </tbody> </table>	IN	OUT	2.5	3	1.4	1	-2.5	-3	-1.4	-1
Description	Examples																	
In accordance with DIN 1333, REAL data types are rounded. If the first decimal place is equal to or greater than 5, the data type is rounded up, otherwise it is rounded down.	<table border="1"> <thead> <tr> <th>IN</th><th>OUT</th></tr> </thead> <tbody> <tr> <td>2.5</td><td>3</td></tr> <tr> <td>1.4</td><td>1</td></tr> <tr> <td>-2.5</td><td>-3</td></tr> <tr> <td>-1.4</td><td>-1</td></tr> </tbody> </table>	IN	OUT	2.5	3	1.4	1	-2.5	-3	-1.4	-1							
IN	OUT																	
2.5	3																	
1.4	1																	
-2.5	-3																	
-1.4	-1																	
OUT	SINT	<p>Whenever the rounded value of IN is greater than $2^{31} - 1$ (16#7FFF FFFF), -1 is output.</p> <p>Whenever the rounded value of IN is less than -2^{31} 0 is output.</p>	<table border="1"> <thead> <tr> <th>IN</th><th>OUT</th></tr> </thead> <tbody> <tr> <td>2147483647.0</td><td>-1</td></tr> <tr> <td>-2147483647.0</td><td>0</td></tr> <tr> <td>-3.66455e+18</td><td>0</td></tr> </tbody> </table>	IN	OUT	2147483647.0	-1	-2147483647.0	0	-3.66455e+18	0							
IN	OUT																	
2147483647.0	-1																	
-2147483647.0	0																	
-3.66455e+18	0																	
		Input data type: TIME																
		<table border="1"> <thead> <tr> <th>Description</th><th>Examples</th></tr> </thead> <tbody> <tr> <td>If the IN data type is TIME, the value of IN is interpreted in milliseconds [ms] and converted to SINT.</td><td> <table border="1"> <thead> <tr> <th>IN</th><th>OUT</th></tr> </thead> <tbody> <tr> <td>T#127ms</td><td>127</td></tr> </tbody> </table> </td></tr> </tbody> </table>			Description	Examples	If the IN data type is TIME, the value of IN is interpreted in milliseconds [ms] and converted to SINT.	<table border="1"> <thead> <tr> <th>IN</th><th>OUT</th></tr> </thead> <tbody> <tr> <td>T#127ms</td><td>127</td></tr> </tbody> </table>	IN	OUT	T#127ms	127						
Description	Examples																	
If the IN data type is TIME, the value of IN is interpreted in milliseconds [ms] and converted to SINT.	<table border="1"> <thead> <tr> <th>IN</th><th>OUT</th></tr> </thead> <tbody> <tr> <td>T#127ms</td><td>127</td></tr> </tbody> </table>	IN	OUT	T#127ms	127													
IN	OUT																	
T#127ms	127																	
<u>ENO</u>	BOOL	TRUE	If EN = TRUE and the function was processed without errors.															
		FALSE	<ul style="list-style-type: none"> • If also EN = FALSE or • Error: The result is out of the range of values for SINT. 															

5.8.3 Conversion Within the Ranges of Values

Data type at IN	Input lower limit	Output lower limit	Input upper limit	Output upper limit
BOOL	0	0	1	1
BYTE	16#00	0	16#7F	127
DINT	-128	-128	127	127
DWORD	16#00	0	16#7F	127
INT	-128	-128	127	127
LREAL	-128.0	-128	127.0	127
REAL	-128.0	-128	127.0	127
SINT	-128	-128	127	127
TIME	T#-2m8s	-128	T#2m7s	127
UDINT	0	0	127	127
UINT	0	0	127	127
USINT	0	0	127	127
WORD	16#00	0	16#7F	127

5.8.4 Conversion Outside the Ranges of Values

Data type at IN	Input lower limit - 1	Output lower limit	Input upper limit + 1	Output upper limit
BOOL	-/-	-/-	-/-	-/-
BYTE	-/-	-/-	128	-128
DINT	-129	127	128	-128
DWORD	-/-	-/-	16#80	-128
INT	-129	127	128	-128
LREAL	-129.0	127	128.0	-128
REAL	-129.0	127	128.8	-128
SINT	-/-	-/-	-/-	-/-
TIME	T#-2m9s	127	T#2m8s	-128
UDINT	-/-	-/-	128	-128
UINT	-/-	-/-	128	-128
USINT	-/-	-/-	128	-128
WORD	-/-	-/-	16#80	-128

5.9 ATOUINT (Any to Unsigned INT)

5.9.1 Inputs

Input	Data type	Description				
IN	ANY ELEMENTARY	<p>Input IN</p>				
EN	BOOL	<p>EN is hidden by default. If EN is connected, it is always TRUE.</p> <p>Show EN/ENO: Right-click a function block or function, and select Show EN/ENO from the context menu.</p> <p>As soon as EN or ENO is connected, it can no longer be hidden.</p> <table border="1"> <tr> <td>TRUE</td><td> <p>The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle.</p> <p>ENO Output: The output is TRUE and can be set to FALSE separately based on the internal error detection.</p> </td></tr> <tr> <td>FALSE</td><td> <p>The function block or function is not processed by the user program. The following variables and value field assignment are not assigned.</p> <p>The outputs of a function block display the results of the user program cycle that was lastly processed.</p> <p>The outputs of a function display the default values defined for the data type.</p> <p>As long as the function block or function has not been completely processed, the outputs retain the default values defined for the data type.</p> <p>ENO Output: The output is also set to FALSE.</p> </td></tr> </table>	TRUE	<p>The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle.</p> <p>ENO Output: The output is TRUE and can be set to FALSE separately based on the internal error detection.</p>	FALSE	<p>The function block or function is not processed by the user program. The following variables and value field assignment are not assigned.</p> <p>The outputs of a function block display the results of the user program cycle that was lastly processed.</p> <p>The outputs of a function display the default values defined for the data type.</p> <p>As long as the function block or function has not been completely processed, the outputs retain the default values defined for the data type.</p> <p>ENO Output: The output is also set to FALSE.</p>
TRUE	<p>The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle.</p> <p>ENO Output: The output is TRUE and can be set to FALSE separately based on the internal error detection.</p>					
FALSE	<p>The function block or function is not processed by the user program. The following variables and value field assignment are not assigned.</p> <p>The outputs of a function block display the results of the user program cycle that was lastly processed.</p> <p>The outputs of a function display the default values defined for the data type.</p> <p>As long as the function block or function has not been completely processed, the outputs retain the default values defined for the data type.</p> <p>ENO Output: The output is also set to FALSE.</p>					

5.9.2 Outputs

Output	Data type	Description																																																									
		<p>Converts the data type at IN to UINT.</p> <p>The data type at IN is determined by creating a typecasting line.</p> <p>Input data type: BOOL</p> <table border="1"> <thead> <tr> <th>Description</th><th colspan="2">Examples</th></tr> </thead> <tbody> <tr> <td>The value TRUE and FALSE at IN are output to OUT as 1 and 0, respectively.</td><td>IN</td><td>OUT</td></tr> <tr> <td>TRUE</td><td>1</td><td></td></tr> <tr> <td>FALSE</td><td>0</td><td></td></tr> </tbody> </table> <p>Input data type: ANY_INT</p> <table border="1"> <thead> <tr> <th>Description</th><th colspan="2">Examples</th></tr> </thead> <tbody> <tr> <td>If IN is greater than the value that can be represented with UINT, IN is limited to [IN modulus 2^{16}] and ENO is FALSE. If IN is negative, the corresponding two's complement is output to OUT and ENO is FALSE.</td><td>IN</td><td>OUT</td></tr> <tr> <td>$2^{16} + 5$</td><td>5</td><td></td></tr> <tr> <td>$2^{63} - 1$</td><td>65535</td><td></td></tr> <tr> <td>-1</td><td>65535</td><td></td></tr> <tr> <td>$-(2^{32} - 1)$</td><td>1</td><td></td></tr> </tbody> </table> <p>Input data type: ANY_REAL</p> <table border="1"> <thead> <tr> <th>Description</th><th colspan="2">Examples</th></tr> </thead> <tbody> <tr> <td>In accordance with DIN 1333, REAL data types are rounded. If the first decimal place is equal to or greater than 5, the data type is rounded up, otherwise it is rounded down. If IN is negative, the corresponding two's complement is output to OUT and ENO is FALSE.</td><td>IN</td><td>OUT</td></tr> <tr> <td>2.5</td><td>3</td><td></td></tr> <tr> <td>1.4</td><td>1</td><td></td></tr> <tr> <td>-2.5</td><td>65533</td><td></td></tr> <tr> <td>-1.4</td><td>65535</td><td></td></tr> <tr> <td>If the rounded value of IN is greater than $2^{32} - 1$, OUT contains the largest value that can be represented with UINT and ENO is FALSE. If the rounded value of IN is less than -2^{31}, OUT contains the lowest value that can be represented with UINT and ENO is FALSE.</td><td>IN</td><td>OUT</td></tr> <tr> <td>1.4e+18</td><td>65535</td><td></td></tr> <tr> <td>-1.4e+18</td><td>0</td><td></td></tr> </tbody> </table>	Description	Examples		The value TRUE and FALSE at IN are output to OUT as 1 and 0, respectively.	IN	OUT	TRUE	1		FALSE	0		Description	Examples		If IN is greater than the value that can be represented with UINT, IN is limited to [IN modulus 2^{16}] and ENO is FALSE. If IN is negative, the corresponding two's complement is output to OUT and ENO is FALSE.	IN	OUT	$2^{16} + 5$	5		$2^{63} - 1$	65535		-1	65535		$-(2^{32} - 1)$	1		Description	Examples		In accordance with DIN 1333, REAL data types are rounded. If the first decimal place is equal to or greater than 5, the data type is rounded up, otherwise it is rounded down. If IN is negative, the corresponding two's complement is output to OUT and ENO is FALSE.	IN	OUT	2.5	3		1.4	1		-2.5	65533		-1.4	65535		If the rounded value of IN is greater than $2^{32} - 1$, OUT contains the largest value that can be represented with UINT and ENO is FALSE. If the rounded value of IN is less than -2^{31} , OUT contains the lowest value that can be represented with UINT and ENO is FALSE.	IN	OUT	1.4e+18	65535		-1.4e+18	0	
Description	Examples																																																										
The value TRUE and FALSE at IN are output to OUT as 1 and 0, respectively.	IN	OUT																																																									
TRUE	1																																																										
FALSE	0																																																										
Description	Examples																																																										
If IN is greater than the value that can be represented with UINT, IN is limited to [IN modulus 2^{16}] and ENO is FALSE. If IN is negative, the corresponding two's complement is output to OUT and ENO is FALSE.	IN	OUT																																																									
$2^{16} + 5$	5																																																										
$2^{63} - 1$	65535																																																										
-1	65535																																																										
$-(2^{32} - 1)$	1																																																										
Description	Examples																																																										
In accordance with DIN 1333, REAL data types are rounded. If the first decimal place is equal to or greater than 5, the data type is rounded up, otherwise it is rounded down. If IN is negative, the corresponding two's complement is output to OUT and ENO is FALSE.	IN	OUT																																																									
2.5	3																																																										
1.4	1																																																										
-2.5	65533																																																										
-1.4	65535																																																										
If the rounded value of IN is greater than $2^{32} - 1$, OUT contains the largest value that can be represented with UINT and ENO is FALSE. If the rounded value of IN is less than -2^{31} , OUT contains the lowest value that can be represented with UINT and ENO is FALSE.	IN	OUT																																																									
1.4e+18	65535																																																										
-1.4e+18	0																																																										

Output	Data type	Description																	
OUT	UINT	Input data type: TIME <table border="1"> <thead> <tr> <th>Description</th><th>Examples</th></tr> </thead> <tbody> <tr> <td>IN must be within -2^{63} ms and $(2^{63} - 1)$ ms. If IN is less than 2^{16} the value is interpreted in milliseconds [ms] and converted to UINT, and ENO is TRUE. If IN is greater than $2^{16} - 1$, IN is limited to [IN modulus 2^{16}] and ENO is FALSE. If IN is negative, the two's complement is output to OUT and ENO is FALSE.</td><td> <table border="1"> <thead> <tr> <th>IN</th><th>OUT</th></tr> </thead> <tbody> <tr> <td>T#1m5s534ms</td><td>65534</td></tr> <tr> <td>T#49d17h2m47s295ms</td><td>65535</td></tr> <tr> <td>T#106751991167d▼ 7h12m55s807ms</td><td>65535</td></tr> <tr> <td>T#-1ms</td><td>65535</td></tr> <tr> <td>T#-49d17h2m▼ 47s295ms</td><td>1</td></tr> </tbody> </table> </td></tr> </tbody> </table>		Description	Examples	IN must be within -2^{63} ms and $(2^{63} - 1)$ ms. If IN is less than 2^{16} the value is interpreted in milliseconds [ms] and converted to UINT, and ENO is TRUE. If IN is greater than $2^{16} - 1$, IN is limited to [IN modulus 2^{16}] and ENO is FALSE. If IN is negative, the two's complement is output to OUT and ENO is FALSE.	<table border="1"> <thead> <tr> <th>IN</th><th>OUT</th></tr> </thead> <tbody> <tr> <td>T#1m5s534ms</td><td>65534</td></tr> <tr> <td>T#49d17h2m47s295ms</td><td>65535</td></tr> <tr> <td>T#106751991167d▼ 7h12m55s807ms</td><td>65535</td></tr> <tr> <td>T#-1ms</td><td>65535</td></tr> <tr> <td>T#-49d17h2m▼ 47s295ms</td><td>1</td></tr> </tbody> </table>	IN	OUT	T#1m5s534ms	65534	T#49d17h2m47s295ms	65535	T#106751991167d▼ 7h12m55s807ms	65535	T#-1ms	65535	T#-49d17h2m▼ 47s295ms	1
Description	Examples																		
IN must be within -2^{63} ms and $(2^{63} - 1)$ ms. If IN is less than 2^{16} the value is interpreted in milliseconds [ms] and converted to UINT, and ENO is TRUE. If IN is greater than $2^{16} - 1$, IN is limited to [IN modulus 2^{16}] and ENO is FALSE. If IN is negative, the two's complement is output to OUT and ENO is FALSE.	<table border="1"> <thead> <tr> <th>IN</th><th>OUT</th></tr> </thead> <tbody> <tr> <td>T#1m5s534ms</td><td>65534</td></tr> <tr> <td>T#49d17h2m47s295ms</td><td>65535</td></tr> <tr> <td>T#106751991167d▼ 7h12m55s807ms</td><td>65535</td></tr> <tr> <td>T#-1ms</td><td>65535</td></tr> <tr> <td>T#-49d17h2m▼ 47s295ms</td><td>1</td></tr> </tbody> </table>	IN	OUT	T#1m5s534ms	65534	T#49d17h2m47s295ms	65535	T#106751991167d▼ 7h12m55s807ms	65535	T#-1ms	65535	T#-49d17h2m▼ 47s295ms	1						
IN	OUT																		
T#1m5s534ms	65534																		
T#49d17h2m47s295ms	65535																		
T#106751991167d▼ 7h12m55s807ms	65535																		
T#-1ms	65535																		
T#-49d17h2m▼ 47s295ms	1																		
<u>ENO</u>	BOOL	<table border="1"> <tr> <td>TRUE</td><td>If EN = TRUE and the function was processed without errors.</td></tr> <tr> <td>FALSE</td><td> <ul style="list-style-type: none"> If also EN = FALSE or Error: The result is out of the range of values for the UINT data type. </td></tr> </table>	TRUE	If EN = TRUE and the function was processed without errors.	FALSE	<ul style="list-style-type: none"> If also EN = FALSE or Error: The result is out of the range of values for the UINT data type. 													
TRUE	If EN = TRUE and the function was processed without errors.																		
FALSE	<ul style="list-style-type: none"> If also EN = FALSE or Error: The result is out of the range of values for the UINT data type. 																		

5.9.3 Conversion Within the Ranges of Values

Data type at IN	Input lower limit	Output lower limit	Input upper limit	Output upper limit
BOOL	0	0	1	1
BYTE	16#00	0	16#FF	255
DINT	0	0	65535	65535
DWORD	16#00	0	16#FFFF	65535
INT	0	0	65535	65535
LREAL	0.0	0	65535.0	65535
REAL	0.0	0	65535.0	65535
SINT	0	0	127	127
TIME	T#0ms	0	T#1m5s535ms	65535
UDINT	0	0	65535	65535
UINT	0	0	65535	65535
USINT	0	0	255	255
WORD	16#00	0	16#FFFF	65535

5.9.4 Conversion Outside the Ranges of Values

Data type at IN	Input lower limit - 1	Output lower limit	Input upper limit + 1	Output upper limit
BOOL	-/-	-/-	-/-	-/-
BYTE	-/-	-/-	-/-	-/-
DINT	-1	65535	65536	0
DWORD	-/-	-/-	16#010000	0
INT	-1	65535	-/-	-/-
LREAL	-1.0	65535	65536.0	0
REAL	-1.0	65535	65536.0	0
SINT	-1	65535	-/-	-/-
TIME	T#-1ms	65535	T#1m5s536ms	0
UDINT	-/-	-/-	65536	0
UINT	-/-	-/-	-/-	-/-
USINT	-/-	-/-	-/-	-/-
WORD	-/-	-/-	-/-	-/-

5.10 ATOINT (Any to INT)

5.10.1 Inputs

Input	Data type	Description				
IN	ANY ELEMENTARY	Input IN				
<u>EN</u>	BOOL	<p>EN is hidden by default. If EN is connected, it is always TRUE.</p> <p>Show EN/ENO: Right-click a function block or function, and select Show EN/ENO from the context menu.</p> <p>As soon as EN or ENO is connected, it can no longer be hidden.</p> <table border="1"> <tr> <td>TRUE</td><td>The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle. ENO Output: The output is TRUE and can be set to FALSE separately based on the internal error detection.</td></tr> <tr> <td>FALSE</td><td>The function block or function is not processed by the user program. The following variables and value field assignment are not assigned. The outputs of a function block display the results of the user program cycle that was lastly processed. The outputs of a function display the default values defined for the data type. As long as the function block or function has not been completely processed, the outputs retain the default values defined for the data type. ENO Output: The output is also set to FALSE.</td></tr> </table>	TRUE	The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle. ENO Output: The output is TRUE and can be set to FALSE separately based on the internal error detection.	FALSE	The function block or function is not processed by the user program. The following variables and value field assignment are not assigned. The outputs of a function block display the results of the user program cycle that was lastly processed. The outputs of a function display the default values defined for the data type. As long as the function block or function has not been completely processed, the outputs retain the default values defined for the data type. ENO Output: The output is also set to FALSE.
TRUE	The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle. ENO Output: The output is TRUE and can be set to FALSE separately based on the internal error detection.					
FALSE	The function block or function is not processed by the user program. The following variables and value field assignment are not assigned. The outputs of a function block display the results of the user program cycle that was lastly processed. The outputs of a function display the default values defined for the data type. As long as the function block or function has not been completely processed, the outputs retain the default values defined for the data type. ENO Output: The output is also set to FALSE.					

5.10.2 Outputs

Output	Data type	Description																																														
OUT	INT	<p>Converts the data type at IN to INT. The data type at IN is determined by creating a typecasting line. The following conditions apply to the value of OUT.</p> <p>Input data type: BOOL</p> <table border="1"> <thead> <tr> <th>Description</th><th>Examples</th></tr> </thead> <tbody> <tr> <td>The value TRUE and FALSE at IN are output to OUT as 1 and 0, respectively.</td><td> <table border="1"> <thead> <tr> <th>IN</th><th>OUT</th></tr> </thead> <tbody> <tr> <td>TRUE</td><td>1</td></tr> <tr> <td>FALSE</td><td>0</td></tr> </tbody> </table> </td></tr> </tbody> </table> <p>Input data type: ANY_INT</p> <table border="1"> <thead> <tr> <th>Description</th><th>Examples</th></tr> </thead> <tbody> <tr> <td>If the value of IN exceeds the range of values defined for INT, the portion that can be represented with the INT data type is output to OUT, and ENO is set to FALSE. If IN is positive and exceeds the upper range limit of INT, the corresponding two's complement is output to OUT. If IN is negative and exceeds the lower range limit of INT, the corresponding two's complement is output to OUT.</td><td> <table border="1"> <thead> <tr> <th>IN</th><th>OUT</th></tr> </thead> <tbody> <tr> <td>32768</td><td>-32767</td></tr> <tr> <td>65535</td><td>-1</td></tr> <tr> <td>-32769</td><td>32767</td></tr> <tr> <td>-65536</td><td>0</td></tr> </tbody> </table> </td></tr> </tbody> </table> <p>Input data type: ANY_REAL</p> <table border="1"> <thead> <tr> <th>Description</th><th>Examples</th></tr> </thead> <tbody> <tr> <td>In accordance with DIN 1333, REAL data types are rounded. If the first decimal place is equal to or greater than 5, the data type is rounded up, otherwise it is rounded down.</td><td> <table border="1"> <thead> <tr> <th>IN</th><th>OUT</th></tr> </thead> <tbody> <tr> <td>2.5</td><td>3</td></tr> <tr> <td>1.4</td><td>1</td></tr> <tr> <td>-2.5</td><td>-3</td></tr> <tr> <td>-1.4</td><td>-1</td></tr> </tbody> </table> </td></tr> <tr> <td>Whenever the rounded value of IN is greater than $2^{31} - 1$ (16#7FFF FFFF), -1 is output. Whenever the rounded value of IN is less than -2^{31} 0 is output.</td><td> <table border="1"> <thead> <tr> <th>IN</th><th>OUT</th></tr> </thead> <tbody> <tr> <td>2147483647.0</td><td>-1</td></tr> <tr> <td>-2147483647.0</td><td>0</td></tr> </tbody> </table> </td></tr> </tbody> </table>	Description	Examples	The value TRUE and FALSE at IN are output to OUT as 1 and 0, respectively.	<table border="1"> <thead> <tr> <th>IN</th><th>OUT</th></tr> </thead> <tbody> <tr> <td>TRUE</td><td>1</td></tr> <tr> <td>FALSE</td><td>0</td></tr> </tbody> </table>	IN	OUT	TRUE	1	FALSE	0	Description	Examples	If the value of IN exceeds the range of values defined for INT, the portion that can be represented with the INT data type is output to OUT, and ENO is set to FALSE. If IN is positive and exceeds the upper range limit of INT, the corresponding two's complement is output to OUT. If IN is negative and exceeds the lower range limit of INT, the corresponding two's complement is output to OUT.	<table border="1"> <thead> <tr> <th>IN</th><th>OUT</th></tr> </thead> <tbody> <tr> <td>32768</td><td>-32767</td></tr> <tr> <td>65535</td><td>-1</td></tr> <tr> <td>-32769</td><td>32767</td></tr> <tr> <td>-65536</td><td>0</td></tr> </tbody> </table>	IN	OUT	32768	-32767	65535	-1	-32769	32767	-65536	0	Description	Examples	In accordance with DIN 1333, REAL data types are rounded. If the first decimal place is equal to or greater than 5, the data type is rounded up, otherwise it is rounded down.	<table border="1"> <thead> <tr> <th>IN</th><th>OUT</th></tr> </thead> <tbody> <tr> <td>2.5</td><td>3</td></tr> <tr> <td>1.4</td><td>1</td></tr> <tr> <td>-2.5</td><td>-3</td></tr> <tr> <td>-1.4</td><td>-1</td></tr> </tbody> </table>	IN	OUT	2.5	3	1.4	1	-2.5	-3	-1.4	-1	Whenever the rounded value of IN is greater than $2^{31} - 1$ (16#7FFF FFFF), -1 is output. Whenever the rounded value of IN is less than -2^{31} 0 is output.	<table border="1"> <thead> <tr> <th>IN</th><th>OUT</th></tr> </thead> <tbody> <tr> <td>2147483647.0</td><td>-1</td></tr> <tr> <td>-2147483647.0</td><td>0</td></tr> </tbody> </table>	IN	OUT	2147483647.0	-1	-2147483647.0	0
Description	Examples																																															
The value TRUE and FALSE at IN are output to OUT as 1 and 0, respectively.	<table border="1"> <thead> <tr> <th>IN</th><th>OUT</th></tr> </thead> <tbody> <tr> <td>TRUE</td><td>1</td></tr> <tr> <td>FALSE</td><td>0</td></tr> </tbody> </table>	IN	OUT	TRUE	1	FALSE	0																																									
IN	OUT																																															
TRUE	1																																															
FALSE	0																																															
Description	Examples																																															
If the value of IN exceeds the range of values defined for INT, the portion that can be represented with the INT data type is output to OUT, and ENO is set to FALSE. If IN is positive and exceeds the upper range limit of INT, the corresponding two's complement is output to OUT. If IN is negative and exceeds the lower range limit of INT, the corresponding two's complement is output to OUT.	<table border="1"> <thead> <tr> <th>IN</th><th>OUT</th></tr> </thead> <tbody> <tr> <td>32768</td><td>-32767</td></tr> <tr> <td>65535</td><td>-1</td></tr> <tr> <td>-32769</td><td>32767</td></tr> <tr> <td>-65536</td><td>0</td></tr> </tbody> </table>	IN	OUT	32768	-32767	65535	-1	-32769	32767	-65536	0																																					
IN	OUT																																															
32768	-32767																																															
65535	-1																																															
-32769	32767																																															
-65536	0																																															
Description	Examples																																															
In accordance with DIN 1333, REAL data types are rounded. If the first decimal place is equal to or greater than 5, the data type is rounded up, otherwise it is rounded down.	<table border="1"> <thead> <tr> <th>IN</th><th>OUT</th></tr> </thead> <tbody> <tr> <td>2.5</td><td>3</td></tr> <tr> <td>1.4</td><td>1</td></tr> <tr> <td>-2.5</td><td>-3</td></tr> <tr> <td>-1.4</td><td>-1</td></tr> </tbody> </table>	IN	OUT	2.5	3	1.4	1	-2.5	-3	-1.4	-1																																					
IN	OUT																																															
2.5	3																																															
1.4	1																																															
-2.5	-3																																															
-1.4	-1																																															
Whenever the rounded value of IN is greater than $2^{31} - 1$ (16#7FFF FFFF), -1 is output. Whenever the rounded value of IN is less than -2^{31} 0 is output.	<table border="1"> <thead> <tr> <th>IN</th><th>OUT</th></tr> </thead> <tbody> <tr> <td>2147483647.0</td><td>-1</td></tr> <tr> <td>-2147483647.0</td><td>0</td></tr> </tbody> </table>	IN	OUT	2147483647.0	-1	-2147483647.0	0																																									
IN	OUT																																															
2147483647.0	-1																																															
-2147483647.0	0																																															

Output	Data type	Description											
OUT	INT	Input data type: TIME											
		<table border="1"> <thead> <tr> <th>Description</th><th colspan="2">Examples</th></tr> </thead> <tbody> <tr> <td>If the IN data type is TIME, the value of IN is interpreted in milliseconds [ms] and converted to INT.</td><td>IN</td><td>OUT</td></tr> <tr> <td></td><td>T#127ms</td><td>127</td></tr> </tbody> </table>			Description	Examples		If the IN data type is TIME, the value of IN is interpreted in milliseconds [ms] and converted to INT.	IN	OUT		T#127ms	127
Description	Examples												
If the IN data type is TIME, the value of IN is interpreted in milliseconds [ms] and converted to INT.	IN	OUT											
	T#127ms	127											
ENO	BOOL	TRUE	If EN = TRUE and the function was processed without errors.										
		FALSE	<ul style="list-style-type: none"> • If also EN = FALSE or • Error: The result is out of the range of values for INT. 										

5.10.3 Conversion Within the Ranges of Values

Data type at IN	Input lower limit	Output lower limit	Input upper limit	Output upper limit
BOOL	0	0	1	1
BYTE	16#00	0	16#FF	255
DINT	-32768	-32768	32767	32767
DWORD	16#00	0	16#7FFF	32767
INT	-32768	-32768	32767	32767
LREAL	-32768.0	-32768	32767.0	32767
REAL	-32768.0	-32768	32767.0	32767
SINT	-128	-128	127	127
TIME	T#-32s768ms	-32768	T#32s767ms	32767
UDINT	0	0	32767	32767
UINT	0	0	32767	32767
USINT	0	0	255	255
WORD	16#00	0	16#7FFF	32767

5.10.4 Conversion Outside the Ranges of Values

Data type at IN	Input lower limit - 1	Output lower limit	Input upper limit + 1	Output upper limit
BOOL	-/-	-/-	-/-	-/-
BYTE	-/-	-/-	-/-	-/-
DINT	-32769	32767	32768	-32768
DWORD	-/-	-/-	16#8000	-32768
INT	-/-	-/-	-/-	-/-
LREAL	-32769.0	32767	32768.0	-32768
REAL	-32769.0	32767	32768.0	-32768
SINT	-/-	-/-	-/-	-/-
TIME	T#-32s769ms	32767	T#32s768ms	-32768
UDINT	-/-	-/-	32768	-32768
UINT	-/-	-/-	32768	-32768
USINT	-/-	-/-	-/-	-/-
WORD	-/-	-/-	16#8000	-32768

5.11 AToudint (Any to Unsigned Double INT)

5.11.1 Inputs

Input	Data type	Description				
IN	ANY ELEMENTARY	Input IN				
<u>EN</u>	BOOL	<p>EN is hidden by default. If EN is connected, it is always TRUE.</p> <p>Show EN/ENO: Right-click a function block or function, and select Show EN/ENO from the context menu.</p> <p>As soon as EN or ENO is connected, it can no longer be hidden.</p> <table border="1"> <tr> <td>TRUE</td><td>The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle. ENO Output: The output is TRUE and can be set to FALSE separately based on the internal error detection.</td></tr> <tr> <td>FALSE</td><td>The function block or function is not processed by the user program. The following variables and value field assignment are not assigned. The outputs of a function block display the results of the user program cycle that was lastly processed. The outputs of a function display the default values defined for the data type. As long as the function block or function has not been completely processed, the outputs retain the default values defined for the data type. ENO Output: The output is also set to FALSE.</td></tr> </table>	TRUE	The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle. ENO Output: The output is TRUE and can be set to FALSE separately based on the internal error detection.	FALSE	The function block or function is not processed by the user program. The following variables and value field assignment are not assigned. The outputs of a function block display the results of the user program cycle that was lastly processed. The outputs of a function display the default values defined for the data type. As long as the function block or function has not been completely processed, the outputs retain the default values defined for the data type. ENO Output: The output is also set to FALSE.
TRUE	The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle. ENO Output: The output is TRUE and can be set to FALSE separately based on the internal error detection.					
FALSE	The function block or function is not processed by the user program. The following variables and value field assignment are not assigned. The outputs of a function block display the results of the user program cycle that was lastly processed. The outputs of a function display the default values defined for the data type. As long as the function block or function has not been completely processed, the outputs retain the default values defined for the data type. ENO Output: The output is also set to FALSE.					

5.11.2 Outputs

Output	Data type	Description																																																									
OUT	UDINT	<p>Converts the data type at IN to UDINT. The data type at IN is determined by creating a typecasting line.</p> <p>Input data type: BOOL</p> <table border="1"> <thead> <tr> <th>Description</th><th colspan="2">Examples</th></tr> </thead> <tbody> <tr> <td>The value TRUE and FALSE at IN are output to OUT as 1 and 0, respectively.</td><td>IN</td><td>OUT</td></tr> <tr> <td>TRUE</td><td>1</td><td></td></tr> <tr> <td>FALSE</td><td>0</td><td></td></tr> </tbody> </table> <p>Input data type: ANY_INT</p> <table border="1"> <thead> <tr> <th>Description</th><th colspan="2">Examples</th></tr> </thead> <tbody> <tr> <td>If IN is greater than the value that can be represented with UDINT, IN is limited to [IN modulus 2^{32}] and ENO is FALSE. If IN is negative, the corresponding two's complement is output to OUT and ENO is FALSE.</td><td>IN</td><td>OUT</td></tr> <tr> <td>$2^{32} + 5$</td><td>5</td><td></td></tr> <tr> <td>$2^{63} - 1$</td><td>4294967295</td><td></td></tr> <tr> <td>-1</td><td>4294967295</td><td></td></tr> <tr> <td>$-(2^{32} - 1)$</td><td>1</td><td></td></tr> </tbody> </table> <p>Input data type: ANY_REAL</p> <table border="1"> <thead> <tr> <th>Description</th><th colspan="2">Examples</th></tr> </thead> <tbody> <tr> <td>In accordance with DIN 1333, REAL data types are rounded. If the first decimal place is equal to or greater than 5, the data type is rounded up, otherwise it is rounded down. If IN is negative, the corresponding two's complement is output to OUT and ENO is FALSE.</td><td>IN</td><td>OUT</td></tr> <tr> <td>2.5</td><td>3</td><td></td></tr> <tr> <td>1.4</td><td>1</td><td></td></tr> <tr> <td>-2.5</td><td>4294967293</td><td></td></tr> <tr> <td>-1.4</td><td>4294967295</td><td></td></tr> <tr> <td>If the rounded value of IN is outside the range of values, OUT contains the largest or lowest possible value and ENO is FALSE.</td><td>IN</td><td>OUT</td></tr> <tr> <td>$1.4e+18$</td><td>4294967295</td><td></td></tr> <tr> <td>$-1.4e+18$</td><td>2147483648</td><td></td></tr> </tbody> </table>	Description	Examples		The value TRUE and FALSE at IN are output to OUT as 1 and 0, respectively.	IN	OUT	TRUE	1		FALSE	0		Description	Examples		If IN is greater than the value that can be represented with UDINT, IN is limited to [IN modulus 2^{32}] and ENO is FALSE. If IN is negative, the corresponding two's complement is output to OUT and ENO is FALSE.	IN	OUT	$2^{32} + 5$	5		$2^{63} - 1$	4294967295		-1	4294967295		$-(2^{32} - 1)$	1		Description	Examples		In accordance with DIN 1333, REAL data types are rounded. If the first decimal place is equal to or greater than 5, the data type is rounded up, otherwise it is rounded down. If IN is negative, the corresponding two's complement is output to OUT and ENO is FALSE.	IN	OUT	2.5	3		1.4	1		-2.5	4294967293		-1.4	4294967295		If the rounded value of IN is outside the range of values, OUT contains the largest or lowest possible value and ENO is FALSE.	IN	OUT	$1.4e+18$	4294967295		$-1.4e+18$	2147483648	
Description	Examples																																																										
The value TRUE and FALSE at IN are output to OUT as 1 and 0, respectively.	IN	OUT																																																									
TRUE	1																																																										
FALSE	0																																																										
Description	Examples																																																										
If IN is greater than the value that can be represented with UDINT, IN is limited to [IN modulus 2^{32}] and ENO is FALSE. If IN is negative, the corresponding two's complement is output to OUT and ENO is FALSE.	IN	OUT																																																									
$2^{32} + 5$	5																																																										
$2^{63} - 1$	4294967295																																																										
-1	4294967295																																																										
$-(2^{32} - 1)$	1																																																										
Description	Examples																																																										
In accordance with DIN 1333, REAL data types are rounded. If the first decimal place is equal to or greater than 5, the data type is rounded up, otherwise it is rounded down. If IN is negative, the corresponding two's complement is output to OUT and ENO is FALSE.	IN	OUT																																																									
2.5	3																																																										
1.4	1																																																										
-2.5	4294967293																																																										
-1.4	4294967295																																																										
If the rounded value of IN is outside the range of values, OUT contains the largest or lowest possible value and ENO is FALSE.	IN	OUT																																																									
$1.4e+18$	4294967295																																																										
$-1.4e+18$	2147483648																																																										

Output	Data type	Description																	
OUT	UDINT	Input data type: TIME <table border="1"> <thead> <tr> <th>Description</th><th>Examples</th></tr> </thead> <tbody> <tr> <td>IN must be within -2^{63} ms and $(2^{63} - 1)$ ms. If IN is less than 2^{32} the value is interpreted in milliseconds [ms] and converted to UDINT. If IN is greater than $2^{32} - 1$, IN is limited to [IN modulus 2^{32}] and ENO is FALSE. If IN is negative, the two's complement is output to OUT and ENO is FALSE.</td><td> <table border="1"> <thead> <tr> <th>IN</th><th>OUT</th></tr> </thead> <tbody> <tr> <td>T#3s</td><td>3000</td></tr> <tr> <td>T#49d17h2m47s295ms</td><td>4294967295</td></tr> <tr> <td>T#106751991167d▼ 7h12m55s807ms</td><td>4294967295</td></tr> <tr> <td>T#-1ms</td><td>4294967295</td></tr> <tr> <td>T#-49d17h2m47s295ms</td><td>1</td></tr> </tbody> </table> </td></tr> </tbody> </table>		Description	Examples	IN must be within -2^{63} ms and $(2^{63} - 1)$ ms. If IN is less than 2^{32} the value is interpreted in milliseconds [ms] and converted to UDINT. If IN is greater than $2^{32} - 1$, IN is limited to [IN modulus 2^{32}] and ENO is FALSE. If IN is negative, the two's complement is output to OUT and ENO is FALSE.	<table border="1"> <thead> <tr> <th>IN</th><th>OUT</th></tr> </thead> <tbody> <tr> <td>T#3s</td><td>3000</td></tr> <tr> <td>T#49d17h2m47s295ms</td><td>4294967295</td></tr> <tr> <td>T#106751991167d▼ 7h12m55s807ms</td><td>4294967295</td></tr> <tr> <td>T#-1ms</td><td>4294967295</td></tr> <tr> <td>T#-49d17h2m47s295ms</td><td>1</td></tr> </tbody> </table>	IN	OUT	T#3s	3000	T#49d17h2m47s295ms	4294967295	T#106751991167d▼ 7h12m55s807ms	4294967295	T#-1ms	4294967295	T#-49d17h2m47s295ms	1
Description	Examples																		
IN must be within -2^{63} ms and $(2^{63} - 1)$ ms. If IN is less than 2^{32} the value is interpreted in milliseconds [ms] and converted to UDINT. If IN is greater than $2^{32} - 1$, IN is limited to [IN modulus 2^{32}] and ENO is FALSE. If IN is negative, the two's complement is output to OUT and ENO is FALSE.	<table border="1"> <thead> <tr> <th>IN</th><th>OUT</th></tr> </thead> <tbody> <tr> <td>T#3s</td><td>3000</td></tr> <tr> <td>T#49d17h2m47s295ms</td><td>4294967295</td></tr> <tr> <td>T#106751991167d▼ 7h12m55s807ms</td><td>4294967295</td></tr> <tr> <td>T#-1ms</td><td>4294967295</td></tr> <tr> <td>T#-49d17h2m47s295ms</td><td>1</td></tr> </tbody> </table>	IN	OUT	T#3s	3000	T#49d17h2m47s295ms	4294967295	T#106751991167d▼ 7h12m55s807ms	4294967295	T#-1ms	4294967295	T#-49d17h2m47s295ms	1						
IN	OUT																		
T#3s	3000																		
T#49d17h2m47s295ms	4294967295																		
T#106751991167d▼ 7h12m55s807ms	4294967295																		
T#-1ms	4294967295																		
T#-49d17h2m47s295ms	1																		
<u>ENO</u>	BOOL	<table border="1"> <tr> <td>TRUE</td><td>If EN = TRUE and the function was processed without errors.</td></tr> <tr> <td>FALSE</td><td> <ul style="list-style-type: none"> If also EN = FALSE or Error: The result is out of the range of values for the UDINT data type. </td></tr> </table>	TRUE	If EN = TRUE and the function was processed without errors.	FALSE	<ul style="list-style-type: none"> If also EN = FALSE or Error: The result is out of the range of values for the UDINT data type. 													
TRUE	If EN = TRUE and the function was processed without errors.																		
FALSE	<ul style="list-style-type: none"> If also EN = FALSE or Error: The result is out of the range of values for the UDINT data type. 																		

5.11.3 Conversion Within the Ranges of Values

Data type at IN	Input lower limit	Output lower limit	Input upper limit	Output upper limit
BOOL	0	0	1	1
BYTE	16#00	0	16#FF	255
DINT	0	0	2147483647	2147483647
DWORD	16#00	0	16#FFFFFFFF	4294967295
INT	0	0	32767	32767
LREAL	0.0	0	4294967295.0	4294967295
REAL	0.0	0	4294967295.0	4294967295
SINT	0	0	127	127
TIME	T#0ms	0	T#49d17h2m47s▼ 295ms	4294967295
UDINT	0	0	4294967295	4294967295
UINT	0	0	65535	65535
USINT	0	0	255	255
WORD	16#00	0	16#FFFF	65535

5.11.4 Conversion Outside the Ranges of Values

Data type at IN	Input lower limit - 1	Output lower limit	Input upper limit + 1	Output upper limit
BOOL	-/-	-/-	-/-	-/-
BYTE	-/-	-/-	-/-	-/-
DINT	-1	4294967295	-/-	-/-
DWORD	-/-	-/-	-/-	-/-
INT	-1	4294967295	-/-	-/-
LREAL	-1.0	4294967295	4294967296.0	4294967295
REAL	-1.0	4294967295	4294967296.0	4294967295
SINT	-1	4294967295	-/-	-/-
TIME	T#-1ms	4294967295	T#49d17h2m47s▼ 296ms	0
UDINT	-/-	-/-	-/-	-/-
UINT	-/-	-/-	-/-	-/-
USINT	-/-	-/-	-/-	-/-
WORD	-/-	-/-	-/-	-/-

5.12 ATODINT (Any to Double INT)

5.12.1 Inputs

Input	Data type	Description				
IN	ANY	<p>Input IN</p>				
<u>EN</u>	BOOL	<p>EN is hidden by default. If EN is connected, it is always TRUE.</p> <p>Show EN/ENO: Right-click a function block or function, and select Show EN/ENO from the context menu.</p> <p>As soon as EN or ENO is connected, it can no longer be hidden.</p> <table border="1"> <tr> <td>TRUE</td><td> <p>The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle.</p> <p>ENO Output: The output is TRUE and can be set to FALSE separately based on the internal error detection.</p> </td></tr> <tr> <td>FALSE</td><td> <p>The function block or function is not processed by the user program. The following variables and value field assignment are not assigned.</p> <p>The outputs of a function block display the results of the user program cycle that was lastly processed.</p> <p>The outputs of a function display the default values defined for the data type.</p> <p>As long as the function block or function has not been completely processed, the outputs retain the default values defined for the data type.</p> <p>ENO Output: The output is also set to FALSE.</p> </td></tr> </table>	TRUE	<p>The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle.</p> <p>ENO Output: The output is TRUE and can be set to FALSE separately based on the internal error detection.</p>	FALSE	<p>The function block or function is not processed by the user program. The following variables and value field assignment are not assigned.</p> <p>The outputs of a function block display the results of the user program cycle that was lastly processed.</p> <p>The outputs of a function display the default values defined for the data type.</p> <p>As long as the function block or function has not been completely processed, the outputs retain the default values defined for the data type.</p> <p>ENO Output: The output is also set to FALSE.</p>
TRUE	<p>The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle.</p> <p>ENO Output: The output is TRUE and can be set to FALSE separately based on the internal error detection.</p>					
FALSE	<p>The function block or function is not processed by the user program. The following variables and value field assignment are not assigned.</p> <p>The outputs of a function block display the results of the user program cycle that was lastly processed.</p> <p>The outputs of a function display the default values defined for the data type.</p> <p>As long as the function block or function has not been completely processed, the outputs retain the default values defined for the data type.</p> <p>ENO Output: The output is also set to FALSE.</p>					

5.12.2 Outputs

Output	Data type	Description																							
OUT	DINT	<p>Converts the data type at IN to DINT. The data type at IN is determined by creating a typecasting line. The following conditions apply to the value of OUT.</p> <p>Input data type: BOOL</p> <table border="1"> <thead> <tr> <th>Description</th><th>Examples</th></tr> </thead> <tbody> <tr> <td>The value TRUE and FALSE at IN are output to OUT as 1 and 0, respectively.</td><td> <table border="1"> <thead> <tr> <th>IN</th><th>OUT</th></tr> </thead> <tbody> <tr> <td>TRUE</td><td>1</td></tr> <tr> <td>FALSE</td><td>0</td></tr> </tbody> </table> </td></tr> </tbody> </table>	Description	Examples	The value TRUE and FALSE at IN are output to OUT as 1 and 0, respectively.	<table border="1"> <thead> <tr> <th>IN</th><th>OUT</th></tr> </thead> <tbody> <tr> <td>TRUE</td><td>1</td></tr> <tr> <td>FALSE</td><td>0</td></tr> </tbody> </table>	IN	OUT	TRUE	1	FALSE	0													
Description	Examples																								
The value TRUE and FALSE at IN are output to OUT as 1 and 0, respectively.	<table border="1"> <thead> <tr> <th>IN</th><th>OUT</th></tr> </thead> <tbody> <tr> <td>TRUE</td><td>1</td></tr> <tr> <td>FALSE</td><td>0</td></tr> </tbody> </table>	IN	OUT	TRUE	1	FALSE	0																		
IN	OUT																								
TRUE	1																								
FALSE	0																								
		<p>Input data type: ANY_INT</p> <table border="1"> <thead> <tr> <th>Description</th><th>Examples</th></tr> </thead> <tbody> <tr> <td>If the value at IN exceeds the range of values defined for DINT, the portion that can be represented with the DINT data type is output to OUT, and ENO is set to FALSE. If IN is positive and exceeds the upper range limit of DINT, the corresponding two's complement is output to OUT. If IN is negative and exceeds the lower range limit of DINT, the corresponding two's complement is output to OUT.</td><td> <table border="1"> <thead> <tr> <th>IN</th><th>OUT</th></tr> </thead> <tbody> <tr> <td>16#FFFF</td><td>65535</td></tr> <tr> <td>2^{31}</td><td>-2147483648</td></tr> <tr> <td>$-2^{31} + 1$</td><td>2147483647</td></tr> </tbody> </table> </td></tr> </tbody> </table>	Description	Examples	If the value at IN exceeds the range of values defined for DINT, the portion that can be represented with the DINT data type is output to OUT, and ENO is set to FALSE. If IN is positive and exceeds the upper range limit of DINT, the corresponding two's complement is output to OUT. If IN is negative and exceeds the lower range limit of DINT, the corresponding two's complement is output to OUT.	<table border="1"> <thead> <tr> <th>IN</th><th>OUT</th></tr> </thead> <tbody> <tr> <td>16#FFFF</td><td>65535</td></tr> <tr> <td>2^{31}</td><td>-2147483648</td></tr> <tr> <td>$-2^{31} + 1$</td><td>2147483647</td></tr> </tbody> </table>	IN	OUT	16#FFFF	65535	2^{31}	-2147483648	$-2^{31} + 1$	2147483647											
Description	Examples																								
If the value at IN exceeds the range of values defined for DINT, the portion that can be represented with the DINT data type is output to OUT, and ENO is set to FALSE. If IN is positive and exceeds the upper range limit of DINT, the corresponding two's complement is output to OUT. If IN is negative and exceeds the lower range limit of DINT, the corresponding two's complement is output to OUT.	<table border="1"> <thead> <tr> <th>IN</th><th>OUT</th></tr> </thead> <tbody> <tr> <td>16#FFFF</td><td>65535</td></tr> <tr> <td>2^{31}</td><td>-2147483648</td></tr> <tr> <td>$-2^{31} + 1$</td><td>2147483647</td></tr> </tbody> </table>	IN	OUT	16#FFFF	65535	2^{31}	-2147483648	$-2^{31} + 1$	2147483647																
IN	OUT																								
16#FFFF	65535																								
2^{31}	-2147483648																								
$-2^{31} + 1$	2147483647																								
		<p>Input data type: ANY_REAL</p> <table border="1"> <thead> <tr> <th>Description</th><th>Examples</th></tr> </thead> <tbody> <tr> <td>In accordance with DIN 1333, REAL data types are rounded. If the first decimal place is equal to or greater than 5, the data type is rounded up, otherwise it is rounded down.</td><td> <table border="1"> <thead> <tr> <th>IN</th><th>OUT</th></tr> </thead> <tbody> <tr> <td>2.5</td><td>3</td></tr> <tr> <td>1.4</td><td>1</td></tr> <tr> <td>-2.5</td><td>-3</td></tr> <tr> <td>-1.4</td><td>-1</td></tr> </tbody> </table> </td></tr> <tr> <td></td><td></td><td> <table border="1"> <thead> <tr> <th>IN</th><th>OUT</th></tr> </thead> <tbody> <tr> <td>$3.0e+9$</td><td>2147483647</td></tr> <tr> <td>$-3.0e+9$</td><td>-2147483648</td></tr> </tbody> </table> </td></tr> </tbody> </table>	Description	Examples	In accordance with DIN 1333, REAL data types are rounded. If the first decimal place is equal to or greater than 5, the data type is rounded up, otherwise it is rounded down.	<table border="1"> <thead> <tr> <th>IN</th><th>OUT</th></tr> </thead> <tbody> <tr> <td>2.5</td><td>3</td></tr> <tr> <td>1.4</td><td>1</td></tr> <tr> <td>-2.5</td><td>-3</td></tr> <tr> <td>-1.4</td><td>-1</td></tr> </tbody> </table>	IN	OUT	2.5	3	1.4	1	-2.5	-3	-1.4	-1			<table border="1"> <thead> <tr> <th>IN</th><th>OUT</th></tr> </thead> <tbody> <tr> <td>$3.0e+9$</td><td>2147483647</td></tr> <tr> <td>$-3.0e+9$</td><td>-2147483648</td></tr> </tbody> </table>	IN	OUT	$3.0e+9$	2147483647	$-3.0e+9$	-2147483648
Description	Examples																								
In accordance with DIN 1333, REAL data types are rounded. If the first decimal place is equal to or greater than 5, the data type is rounded up, otherwise it is rounded down.	<table border="1"> <thead> <tr> <th>IN</th><th>OUT</th></tr> </thead> <tbody> <tr> <td>2.5</td><td>3</td></tr> <tr> <td>1.4</td><td>1</td></tr> <tr> <td>-2.5</td><td>-3</td></tr> <tr> <td>-1.4</td><td>-1</td></tr> </tbody> </table>	IN	OUT	2.5	3	1.4	1	-2.5	-3	-1.4	-1														
IN	OUT																								
2.5	3																								
1.4	1																								
-2.5	-3																								
-1.4	-1																								
		<table border="1"> <thead> <tr> <th>IN</th><th>OUT</th></tr> </thead> <tbody> <tr> <td>$3.0e+9$</td><td>2147483647</td></tr> <tr> <td>$-3.0e+9$</td><td>-2147483648</td></tr> </tbody> </table>	IN	OUT	$3.0e+9$	2147483647	$-3.0e+9$	-2147483648																	
IN	OUT																								
$3.0e+9$	2147483647																								
$-3.0e+9$	-2147483648																								

Output	Data type	Description											
OUT	DINT	Input data type: TIME											
		<table border="1"> <thead> <tr> <th>Description</th><th>Examples</th></tr> </thead> <tbody> <tr> <td>If the IN data type is TIME, the value at IN is interpreted in milliseconds [ms] and converted to DINT.</td><td> <table border="1"> <thead> <tr> <th>IN</th><th>OUT</th></tr> </thead> <tbody> <tr> <td>T#127ms</td><td>127</td></tr> </tbody> </table> </td></tr> </tbody> </table>				Description	Examples	If the IN data type is TIME, the value at IN is interpreted in milliseconds [ms] and converted to DINT.	<table border="1"> <thead> <tr> <th>IN</th><th>OUT</th></tr> </thead> <tbody> <tr> <td>T#127ms</td><td>127</td></tr> </tbody> </table>	IN	OUT	T#127ms	127
Description	Examples												
If the IN data type is TIME, the value at IN is interpreted in milliseconds [ms] and converted to DINT.	<table border="1"> <thead> <tr> <th>IN</th><th>OUT</th></tr> </thead> <tbody> <tr> <td>T#127ms</td><td>127</td></tr> </tbody> </table>	IN	OUT	T#127ms	127								
IN	OUT												
T#127ms	127												
<u>ENO</u>	BOOL	<table border="1"> <tr> <td>TRUE</td><td>If EN = TRUE and the function was processed without errors.</td><td colspan="2" rowspan="2"></td></tr> <tr> <td>FALSE</td><td colspan="3"> <ul style="list-style-type: none"> • If also EN = FALSE or • Error: The result is out of the range of values for DINT. </td></tr> </table>				TRUE	If EN = TRUE and the function was processed without errors.			FALSE	<ul style="list-style-type: none"> • If also EN = FALSE or • Error: The result is out of the range of values for DINT. 		
TRUE	If EN = TRUE and the function was processed without errors.												
FALSE	<ul style="list-style-type: none"> • If also EN = FALSE or • Error: The result is out of the range of values for DINT. 												

5.12.3 Conversion Within the Ranges of Values

Data type at IN	Input lower limit	Output lower limit	Input upper limit	Output upper limit
BOOL	0	0	1	1
BYTE	16#00	0	16#FF	255
DINT	-2147483648	-2147483648	2147483647	2147483647
DWORD	16#00	0	16#7FFFFFFF	2147483647
INT	-32768	-32768	32767	32767
LREAL	-2147483648.0	-2147483648	2147483647.0	2147483647
REAL	-2.14748e+09	-2147480064	2.14748e+09	2147480064
SINT	-128	-128	127	127
TIME	T#-24d20h31m▼ 23s648ms	-2147483648	T#24d20h31m▼ 23s647ms	2147483647
UDINT	0	0	2147483647	2147483647
UINT	0	0	65535	65535
USINT	0	0	255	255
WORD	16#00	0	16#FFFF	65535

5.12.4 Conversion Outside the Ranges of Values

Data type at IN	Input lower limit - 1	Output lower limit	Input upper limit + 1	Output upper limit
BOOL	-/-	-/-	-/-	-/-
BYTE	-/-	-/-	-/-	-/-
DINT	-/-	-/-	-/-	-/-
DWORD	-/-	-/-	16#80000000	-2147483648
INT	-/-	-/-	-/-	-/-
LREAL	-2147483649.0	2147483647	2147483648.0	-2147483648.0
REAL	-2147483649.0	2147483647	2147483648.0	-2147483648.0
SINT	-/-	-/-	-/-	-/-
TIME	-/-	-/-	-/-	-/-
UDINT	-/-	-/-	2147483648	-2147483648
UINT	-/-	-/-	-/-	-/-
USINT	-/-	-/-	-/-	-/-
WORD	-/-	-/-	-/-	-/-

5.13 ATOULINT (Any to Unsigned Long INT)

5.13.1 Inputs

Input	Data type	Description				
IN	ANY ELEMENTARY	Input IN				
EN	BOOL	<p>EN is hidden by default. If EN is connected, it is always TRUE.</p> <p>Show EN/ENO: Right-click a function block or function, and select Show EN/ENO from the context menu.</p> <p>As soon as EN or ENO is connected, it can no longer be hidden.</p> <table border="1"> <tr> <td>TRUE</td><td>The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle. ENO Output: The output is TRUE and can be set to FALSE separately based on the internal error detection.</td></tr> <tr> <td>FALSE</td><td>The function block or function is not processed by the user program. The following variables and value field assignment are not assigned. The outputs of a function block display the results of the user program cycle that was lastly processed. The outputs of a function display the default values defined for the data type. As long as the function block or function has not been completely processed, the outputs retain the default values defined for the data type. ENO Output: The output is also set to FALSE.</td></tr> </table>	TRUE	The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle. ENO Output: The output is TRUE and can be set to FALSE separately based on the internal error detection.	FALSE	The function block or function is not processed by the user program. The following variables and value field assignment are not assigned. The outputs of a function block display the results of the user program cycle that was lastly processed. The outputs of a function display the default values defined for the data type. As long as the function block or function has not been completely processed, the outputs retain the default values defined for the data type. ENO Output: The output is also set to FALSE.
TRUE	The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle. ENO Output: The output is TRUE and can be set to FALSE separately based on the internal error detection.					
FALSE	The function block or function is not processed by the user program. The following variables and value field assignment are not assigned. The outputs of a function block display the results of the user program cycle that was lastly processed. The outputs of a function display the default values defined for the data type. As long as the function block or function has not been completely processed, the outputs retain the default values defined for the data type. ENO Output: The output is also set to FALSE.					

5.13.2 Outputs

Output	Data type	Description																																																						
OUT	ULINT	<p>Converts the data type at IN to ULINT. The data type at IN is determined by creating a typecasting line.</p> <p>Input data type: BOOL</p> <table border="1"> <thead> <tr> <th>Description</th><th colspan="2">Examples</th></tr> </thead> <tbody> <tr> <td>The value TRUE and FALSE at IN are output to OUT as 1 and 0, respectively.</td><td>IN</td><td>OUT</td></tr> <tr> <td></td><td>TRUE</td><td>1</td></tr> <tr> <td></td><td>FALSE</td><td>0</td></tr> </tbody> </table> <p>Input data type: ANY_INT</p> <table border="1"> <thead> <tr> <th>Description</th><th colspan="2">Examples</th></tr> </thead> <tbody> <tr> <td>All positive INT values can be converted to ULINT without restrictions. If IN is negative, the corresponding two's complement is output to OUT and ENO is FALSE.</td><td>IN</td><td>OUT</td></tr> <tr> <td></td><td>$2^{63} - 1$</td><td>9223372036854775807</td></tr> <tr> <td></td><td>-1</td><td>18446744073709551615 (= $2^{64} - 1$)</td></tr> <tr> <td></td><td>$-(2^{63} - 1)$</td><td>9223372036854775809</td></tr> </tbody> </table> <p>Input data type: ANY_REAL</p> <table border="1"> <thead> <tr> <th>Description</th><th colspan="2">Examples</th></tr> </thead> <tbody> <tr> <td>In accordance with DIN 1333, REAL data types are rounded. If the first decimal place is equal to or greater than 5, the data type is rounded up, otherwise it is rounded down. If IN is negative, the corresponding two's complement is output to OUT and ENO is FALSE.</td><td>IN</td><td>OUT</td></tr> <tr> <td></td><td>2.5</td><td>3</td></tr> <tr> <td></td><td>1.4</td><td>1</td></tr> <tr> <td></td><td>-2.5</td><td>18446744073709551613 (= $2^{64} - 3$)</td></tr> <tr> <td></td><td>-1.4</td><td>18446744073709551615 (= $2^{64} - 1$)</td></tr> <tr> <td>If the rounded value of IN is outside the range of values, OUT contains the largest or lowest possible value and ENO is FALSE.</td><td>IN</td><td>OUT</td></tr> <tr> <td></td><td>1.85e+19</td><td>18446744073709551615</td></tr> <tr> <td></td><td>-1.85e+19</td><td>9223372036854775808 (= 2^{63})</td></tr> </tbody> </table>	Description	Examples		The value TRUE and FALSE at IN are output to OUT as 1 and 0, respectively.	IN	OUT		TRUE	1		FALSE	0	Description	Examples		All positive INT values can be converted to ULINT without restrictions. If IN is negative, the corresponding two's complement is output to OUT and ENO is FALSE.	IN	OUT		$2^{63} - 1$	9223372036854775807		-1	18446744073709551615 (= $2^{64} - 1$)		$-(2^{63} - 1)$	9223372036854775809	Description	Examples		In accordance with DIN 1333, REAL data types are rounded. If the first decimal place is equal to or greater than 5, the data type is rounded up, otherwise it is rounded down. If IN is negative, the corresponding two's complement is output to OUT and ENO is FALSE.	IN	OUT		2.5	3		1.4	1		-2.5	18446744073709551613 (= $2^{64} - 3$)		-1.4	18446744073709551615 (= $2^{64} - 1$)	If the rounded value of IN is outside the range of values, OUT contains the largest or lowest possible value and ENO is FALSE.	IN	OUT		1.85e+19	18446744073709551615		-1.85e+19	9223372036854775808 (= 2^{63})
Description	Examples																																																							
The value TRUE and FALSE at IN are output to OUT as 1 and 0, respectively.	IN	OUT																																																						
	TRUE	1																																																						
	FALSE	0																																																						
Description	Examples																																																							
All positive INT values can be converted to ULINT without restrictions. If IN is negative, the corresponding two's complement is output to OUT and ENO is FALSE.	IN	OUT																																																						
	$2^{63} - 1$	9223372036854775807																																																						
	-1	18446744073709551615 (= $2^{64} - 1$)																																																						
	$-(2^{63} - 1)$	9223372036854775809																																																						
Description	Examples																																																							
In accordance with DIN 1333, REAL data types are rounded. If the first decimal place is equal to or greater than 5, the data type is rounded up, otherwise it is rounded down. If IN is negative, the corresponding two's complement is output to OUT and ENO is FALSE.	IN	OUT																																																						
	2.5	3																																																						
	1.4	1																																																						
	-2.5	18446744073709551613 (= $2^{64} - 3$)																																																						
	-1.4	18446744073709551615 (= $2^{64} - 1$)																																																						
If the rounded value of IN is outside the range of values, OUT contains the largest or lowest possible value and ENO is FALSE.	IN	OUT																																																						
	1.85e+19	18446744073709551615																																																						
	-1.85e+19	9223372036854775808 (= 2^{63})																																																						

Output	Data type	Description												
		Input data type: TIME												
OUT	ULINT	Description <p>IN must be within -2^{63} ms and $(2^{63} - 1)$ ms. The value of IN is interpreted in milliseconds [ms] and converted to ULINT. If IN is negative, the two's complement is output to OUT and ENO is FALSE.</p>		Examples										
		<table border="1"> <thead> <tr> <th>IN</th> <th>OUT</th> </tr> </thead> <tbody> <tr> <td>T#3s</td> <td>3000</td> </tr> <tr> <td>T#1m40s</td> <td>10000</td> </tr> <tr> <td>T#106751991167d▼ 7h12m55s807ms</td> <td>9223372036854775807 $(= 2^{63} - 1)$</td> </tr> <tr> <td>T#-1ms</td> <td>18446744073709551615</td> </tr> <tr> <td>T#- 106751991167d▼ 7h12m55s808ms</td> <td>9223372036854775808 $(= 2^{63})$</td> </tr> </tbody> </table>		IN	OUT	T#3s	3000	T#1m40s	10000	T#106751991167d▼ 7h12m55s807ms	9223372036854775807 $(= 2^{63} - 1)$	T#-1ms	18446744073709551615	T#- 106751991167d▼ 7h12m55s808ms
IN	OUT													
T#3s	3000													
T#1m40s	10000													
T#106751991167d▼ 7h12m55s807ms	9223372036854775807 $(= 2^{63} - 1)$													
T#-1ms	18446744073709551615													
T#- 106751991167d▼ 7h12m55s808ms	9223372036854775808 $(= 2^{63})$													
<p>TRUE If EN = TRUE and the function was processed without errors.</p>														
<p>FALSE • If also EN = FALSE or • Error: The result is out of the range of values for the ULINT data type.</p>														
ENO	BOOL													

5.14 ATOLINT (Any to Long INT)

5.14.1 Inputs

Input	Data type	Description				
IN	ANY ELEMENTARY	<p>Input IN</p>				
EN	BOOL	<p>EN is hidden by default. If EN is connected, it is always TRUE.</p> <p>Show EN/ENO: Right-click a function block or function, and select Show EN/ENO from the context menu.</p> <p>As soon as EN or ENO is connected, it can no longer be hidden.</p> <table border="1"> <tr> <td>TRUE</td><td>The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle. ENO Output: The output is TRUE and can be set to FALSE separately based on the internal error detection.</td></tr> <tr> <td>FALSE</td><td>The function block or function is not processed by the user program. The following variables and value field assignment are not assigned. The outputs of a function block display the results of the user program cycle that was lastly processed. The outputs of a function display the default values defined for the data type. As long as the function block or function has not been completely processed, the outputs retain the default values defined for the data type. ENO Output: The output is also set to FALSE.</td></tr> </table>	TRUE	The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle. ENO Output: The output is TRUE and can be set to FALSE separately based on the internal error detection.	FALSE	The function block or function is not processed by the user program. The following variables and value field assignment are not assigned. The outputs of a function block display the results of the user program cycle that was lastly processed. The outputs of a function display the default values defined for the data type. As long as the function block or function has not been completely processed, the outputs retain the default values defined for the data type. ENO Output: The output is also set to FALSE.
TRUE	The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle. ENO Output: The output is TRUE and can be set to FALSE separately based on the internal error detection.					
FALSE	The function block or function is not processed by the user program. The following variables and value field assignment are not assigned. The outputs of a function block display the results of the user program cycle that was lastly processed. The outputs of a function display the default values defined for the data type. As long as the function block or function has not been completely processed, the outputs retain the default values defined for the data type. ENO Output: The output is also set to FALSE.					

5.14.2 Outputs

Output	Data type	Description												
OUT	LINT	<p>Converts a data type at IN to LINT. The data type at IN is determined by creating a typecasting line.</p> <p>The following conditions apply to the value of OUT.</p> <p>Input data type: BOOL</p> <table border="1"> <thead> <tr> <th>Description</th><th colspan="2">Examples</th></tr> <tr> <th></th><th>IN</th><th>OUT</th></tr> </thead> <tbody> <tr> <td>The value TRUE and FALSE at IN are output to OUT as 1 and 0, respectively.</td><td>TRUE</td><td>1</td></tr> <tr> <td></td><td>FALSE</td><td>0</td></tr> </tbody> </table>	Description	Examples			IN	OUT	The value TRUE and FALSE at IN are output to OUT as 1 and 0, respectively.	TRUE	1		FALSE	0
Description	Examples													
	IN	OUT												
The value TRUE and FALSE at IN are output to OUT as 1 and 0, respectively.	TRUE	1												
	FALSE	0												

Output	Data type	Description																				
		Input data type: ANY_INT																				
		<table border="1"> <thead> <tr> <th>Description</th><th colspan="2">Examples</th></tr> </thead> <tbody> <tr> <td>If IN exceeds the range of values for LINT, OUT contains the corresponding two's complement.</td><td>IN</td><td>OUT</td></tr> <tr> <td></td><td>16#FFFF</td><td>65535</td></tr> <tr> <td></td><td>2^{63}</td><td>-2^{63}</td></tr> <tr> <td></td><td>$2^{64} - 1$</td><td>-1</td></tr> </tbody> </table>			Description	Examples		If IN exceeds the range of values for LINT, OUT contains the corresponding two's complement.	IN	OUT		16#FFFF	65535		2^{63}	-2^{63}		$2^{64} - 1$	-1			
Description	Examples																					
If IN exceeds the range of values for LINT, OUT contains the corresponding two's complement.	IN	OUT																				
	16#FFFF	65535																				
	2^{63}	-2^{63}																				
	$2^{64} - 1$	-1																				
		Input data type: ANY_REAL																				
OUT	LINT	<table border="1"> <thead> <tr> <th>Description</th><th colspan="2">Examples</th></tr> </thead> <tbody> <tr> <td>In accordance with DIN 1333, REAL data types are rounded. If the first decimal place is equal to or greater than 5, the data type is rounded up, otherwise it is rounded down.</td><td>IN</td><td>OUT</td></tr> <tr> <td></td><td>2.5</td><td>3</td></tr> <tr> <td></td><td>1.4</td><td>1</td></tr> <tr> <td></td><td>-2.5</td><td>-3</td></tr> <tr> <td></td><td>-1.4</td><td>-1</td></tr> </tbody> </table>			Description	Examples		In accordance with DIN 1333, REAL data types are rounded. If the first decimal place is equal to or greater than 5, the data type is rounded up, otherwise it is rounded down.	IN	OUT		2.5	3		1.4	1		-2.5	-3		-1.4	-1
Description	Examples																					
In accordance with DIN 1333, REAL data types are rounded. If the first decimal place is equal to or greater than 5, the data type is rounded up, otherwise it is rounded down.	IN	OUT																				
	2.5	3																				
	1.4	1																				
	-2.5	-3																				
	-1.4	-1																				
<table border="1"> <tbody> <tr> <td>Whenever the rounded value of IN is greater than $2^{63} - 1$, or lower than -2^{63}, -2^{63} is output. Input values greater or less (+/-) than $9.223372034e+18$ lead to rounding errors.</td><td>IN</td><td>OUT</td></tr> <tr> <td></td><td>$2^{63} + 10$</td><td>-9223372036854▼ 775808</td></tr> <tr> <td></td><td>$-2^{63} + 10$</td><td>-9223372036854▼ 775808</td></tr> </tbody> </table>			Whenever the rounded value of IN is greater than $2^{63} - 1$, or lower than -2^{63} , -2^{63} is output. Input values greater or less (+/-) than $9.223372034e+18$ lead to rounding errors.	IN	OUT		$2^{63} + 10$	-9223372036854▼ 775808		$-2^{63} + 10$	-9223372036854▼ 775808											
Whenever the rounded value of IN is greater than $2^{63} - 1$, or lower than -2^{63} , -2^{63} is output. Input values greater or less (+/-) than $9.223372034e+18$ lead to rounding errors.	IN	OUT																				
	$2^{63} + 10$	-9223372036854▼ 775808																				
	$-2^{63} + 10$	-9223372036854▼ 775808																				
		Input data type: TIME																				
		<table border="1"> <thead> <tr> <th>Description</th><th colspan="2">Examples</th></tr> </thead> <tbody> <tr> <td>If the IN data type is TIME, the value of IN is interpreted in milliseconds [ms] and converted to LINT.</td><td>IN</td><td>OUT</td></tr> <tr> <td></td><td>T#127ms</td><td>127</td></tr> </tbody> </table>			Description	Examples		If the IN data type is TIME, the value of IN is interpreted in milliseconds [ms] and converted to LINT.	IN	OUT		T#127ms	127									
Description	Examples																					
If the IN data type is TIME, the value of IN is interpreted in milliseconds [ms] and converted to LINT.	IN	OUT																				
	T#127ms	127																				
<u>ENO</u>	BOOL	TRUE	If EN = TRUE and the function was processed without errors.																			
		FALSE	<ul style="list-style-type: none"> If also EN = FALSE or Error: The result is out of the range of values for LINT. 																			

5.15 ATOTIME (Any to TIME)

5.15.1 Inputs

Input	Data type	Description				
IN	ANY ELEMENTARY	Input IN				
<u>EN</u>	BOOL	<p>EN is hidden by default. If EN is connected, it is always TRUE.</p> <p>Show EN/ENO: Right-click a function block or function, and select Show EN/ENO from the context menu.</p> <p>As soon as EN or ENO is connected, it can no longer be hidden.</p> <table border="1"> <tr> <td>TRUE</td><td>The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle. ENO Output: The output is TRUE and can be set to FALSE separately based on the internal error detection.</td></tr> <tr> <td>FALSE</td><td>The function block or function is not processed by the user program. The following variables and value field assignment are not assigned. The outputs of a function block display the results of the user program cycle that was lastly processed. The outputs of a function display the default values defined for the data type. As long as the function block or function has not been completely processed, the outputs retain the default values defined for the data type. ENO Output: The output is also set to FALSE.</td></tr> </table>	TRUE	The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle. ENO Output: The output is TRUE and can be set to FALSE separately based on the internal error detection.	FALSE	The function block or function is not processed by the user program. The following variables and value field assignment are not assigned. The outputs of a function block display the results of the user program cycle that was lastly processed. The outputs of a function display the default values defined for the data type. As long as the function block or function has not been completely processed, the outputs retain the default values defined for the data type. ENO Output: The output is also set to FALSE.
TRUE	The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle. ENO Output: The output is TRUE and can be set to FALSE separately based on the internal error detection.					
FALSE	The function block or function is not processed by the user program. The following variables and value field assignment are not assigned. The outputs of a function block display the results of the user program cycle that was lastly processed. The outputs of a function display the default values defined for the data type. As long as the function block or function has not been completely processed, the outputs retain the default values defined for the data type. ENO Output: The output is also set to FALSE.					

5.15.2 Outputs

Output	Data type	Description	
OUT	TIME	<p>Converts the data type at IN1 to TIME (ms).</p> <p>The data type at IN is determined by creating a typecasting line.</p> <p>Only the data types specified in the table below can be converted.</p>	
<u>ENO</u>	BOOL	TRUE	If EN = TRUE and the function was processed without errors.
		FALSE	<ul style="list-style-type: none"> If also EN = FALSE or Error: The result is out of the range of values for TIME.

5.15.3 Conversion Within the Ranges of Values

Data type at IN	Input lower limit	Output lower limit
BOOL		T#0ms
BYTE	16#00	T#0ms
DINT	- 2^{31}	T#-24d20h31m23s648ms
DWORD	16#00	T#0ms
INT	-32768	T#-32s768ms
LREAL	-9.2233720368547758e+18	T#-106751991167d7h12m55s807ms
REAL	-9.2233720368547758e+18	T#-106751991167d7h12m55s807ms
SINT	-128	T#-128ms
TIME	T#-106751991167d7h12m55s807ms	T#-106751991167d7h12m55s807ms
UDINT	0	T#0ms
UINT	0	T#0ms
USINT	0	T#0ms
WORD	16#00	T#0ms

Data type at IN	Input upper limit	Output upper limit
BOOL	1	T#1ms
BYTE	16#FF	T#255ms
DINT	$2^{31} - 1$	T#24d20h31m23s647ms
DWORD	16#FFFF FFFF	T#49d17h2m47s295ms
INT	32767	T#32s767ms
LREAL	9.2233720368547758e+18	T#106751991167d7h12m55s807ms
REAL	9.2233720368547758e+18	T#106751991167d7h12m55s807ms
SINT	127	T#127ms
TIME	T#106751991167d7h12m55s807ms	T#106751991167d7h12m55s807ms
UDINT	$2^{32} - 1$	T#49d17h2m47s295ms
UINT	65535	T#1m5s535ms
USINT	255	T#255ms
WORD	16#FFFF	T#1m5s535ms

5.15.4 Conversion Outside the Ranges of Values

Data type at IN	Input lower limit - 1	Output lower limit
BOOL	-/-	-/-
BYTE	-/-	-/-
DINT	-/-	-/-
DWORD	-/-	-/-
INT	-/-	-/-
LREAL	-9.223372036854775799e+18	T#-106751991167d7h12m55s808ms
REAL	-9.223372036854775799e+18	T#-106751991167d7h12m55s808ms
SINT	-/-	-/-
TIME	-/-	-/-
UDINT	-/-	-/-
UINT	-/-	-/-
USINT	-/-	-/-
WORD	-/-	-/-

Data type at IN	Input upper limit + 1	Output upper limit
BOOL	-/-	-/-
BYTE	-/-	-/-
DINT	-/-	-/-
DWORD	-/-	-/-
INT	-/-	-/-
LREAL	9.223372036854775801e+18	T#-106751991167d7h12m55s808ms
REAL	9.223372036854775801e+18	T#-106751991167d7h12m55s808ms
SINT	-/-	-/-
TIME	-/-	-/-
UDINT	-/-	-/-
UINT	-/-	-/-
USINT	-/-	-/-
WORD	-/-	-/-

5.16 ATOREAL (Any to REAL)

5.16.1 Inputs

Input	Data type	Description				
IN	ANY ELEMENTARY	<p>Input IN</p> <p>Range of values: The absolute value of IN should not exceed 2^{24} to prevent rounding errors.</p>				
<u>EN</u>	BOOL	<p>EN is hidden by default. If EN is connected, it is always TRUE.</p> <p>Show EN/ENO: Right-click a function block or function, and select Show EN/ENO from the context menu.</p> <p>As soon as EN or ENO is connected, it can no longer be hidden.</p> <table border="1"> <tr> <td>TRUE</td><td> <p>The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle.</p> <p>ENO Output: The output is TRUE and can be set to FALSE separately based on the internal error detection.</p> </td></tr> <tr> <td>FALSE</td><td> <p>The function block or function is not processed by the user program. The following variables and value field assignment are not assigned.</p> <p>The outputs of a function block display the results of the user program cycle that was lastly processed.</p> <p>The outputs of a function display the default values defined for the data type.</p> <p>As long as the function block or function has not been completely processed, the outputs retain the default values defined for the data type.</p> <p>ENO Output: The output is also set to FALSE.</p> </td></tr> </table>	TRUE	<p>The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle.</p> <p>ENO Output: The output is TRUE and can be set to FALSE separately based on the internal error detection.</p>	FALSE	<p>The function block or function is not processed by the user program. The following variables and value field assignment are not assigned.</p> <p>The outputs of a function block display the results of the user program cycle that was lastly processed.</p> <p>The outputs of a function display the default values defined for the data type.</p> <p>As long as the function block or function has not been completely processed, the outputs retain the default values defined for the data type.</p> <p>ENO Output: The output is also set to FALSE.</p>
TRUE	<p>The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle.</p> <p>ENO Output: The output is TRUE and can be set to FALSE separately based on the internal error detection.</p>					
FALSE	<p>The function block or function is not processed by the user program. The following variables and value field assignment are not assigned.</p> <p>The outputs of a function block display the results of the user program cycle that was lastly processed.</p> <p>The outputs of a function display the default values defined for the data type.</p> <p>As long as the function block or function has not been completely processed, the outputs retain the default values defined for the data type.</p> <p>ENO Output: The output is also set to FALSE.</p>					

5.16.2 Outputs

Output	Data type	Description								
OUT	REAL	<p>Converts the data type at IN to REAL. The data type at IN is determined by creating a typecasting line.</p> <p>Input data type: DINT, DWORD and greater</p> <table border="1"> <thead> <tr> <th>Description</th><th>Examples</th></tr> </thead> <tbody> <tr> <td>For input values of data types DINT, DWORD and greater, rounding losses can be expected if the modulus of the input values is greater than 2^{24}</td><td> <table border="1"> <thead> <tr> <th>IN</th><th>OUT</th></tr> </thead> <tbody> <tr> <td>2147483645</td><td>2147483650.0 (Wert ist um 5 zu gross)</td></tr> </tbody> </table> </td></tr> </tbody> </table> <p>Also observe the table <i>Converting the Ranges of Values</i>.</p>	Description	Examples	For input values of data types DINT, DWORD and greater, rounding losses can be expected if the modulus of the input values is greater than 2^{24}	<table border="1"> <thead> <tr> <th>IN</th><th>OUT</th></tr> </thead> <tbody> <tr> <td>2147483645</td><td>2147483650.0 (Wert ist um 5 zu gross)</td></tr> </tbody> </table>	IN	OUT	2147483645	2147483650.0 (Wert ist um 5 zu gross)
Description	Examples									
For input values of data types DINT, DWORD and greater, rounding losses can be expected if the modulus of the input values is greater than 2^{24}	<table border="1"> <thead> <tr> <th>IN</th><th>OUT</th></tr> </thead> <tbody> <tr> <td>2147483645</td><td>2147483650.0 (Wert ist um 5 zu gross)</td></tr> </tbody> </table>	IN	OUT	2147483645	2147483650.0 (Wert ist um 5 zu gross)					
IN	OUT									
2147483645	2147483650.0 (Wert ist um 5 zu gross)									
<hr/>										
<u>ENO</u>	BOOL	<table border="1"> <tbody> <tr> <td>TRUE</td><td>If EN = TRUE and the function was processed without errors.</td></tr> <tr> <td>FALSE</td><td> <ul style="list-style-type: none"> • If also EN = FALSE or • Error: IN is out of the range of values for the ANY_REAL data type (+INF, -INF or NaN (not a number)). </td></tr> </tbody> </table>	TRUE	If EN = TRUE and the function was processed without errors.	FALSE	<ul style="list-style-type: none"> • If also EN = FALSE or • Error: IN is out of the range of values for the ANY_REAL data type (+INF, -INF or NaN (not a number)). 				
TRUE	If EN = TRUE and the function was processed without errors.									
FALSE	<ul style="list-style-type: none"> • If also EN = FALSE or • Error: IN is out of the range of values for the ANY_REAL data type (+INF, -INF or NaN (not a number)). 									

5.16.3 Converting the Ranges of Values

Data type at IN	Input lower limit	Output lower limit	Input upper limit	Output upper limit
BOOL	0	0.0	1	1.0
BYTE	16#00	0.0	16#FF	255.0
USINT	0	0.0	255	255.0
UINT	0	0.0	65535	65535.0
UDINT	0	0.0	$2^{32} - 1$	4294967295.0
ULINT	0	0.0	$2^{64} - 1$	1.84467441e+19
INT	-32768	-32768.0	32767	32767.0
SINT	-128	-128.0	127	127.0
DINT	-2^{31}	-2147483648.0	2147483647	2147483647.0
LINT	-2^{63}	-9.22337204e+18	$2^{63} - 1$	9.22337204e+18
WORD	16#0000	0.0	16#FFFF	65535.0
DWORD	16#0000 0000	0.0	16#FFFF FFFF	4294967295.0
LWORD	16#0000 0000 ▼ 0000 0000	0.0	16#7FFF FFFF ▼ FFFF FFFF	9.22337204e+18
REAL	-3.402823e+38	-3.40282306e+38	3.402823e+38	3.40282306e+38
LREAL	-1.7976931348623▼ 158e+308	-INF	1.7976931348623 158e+308	+INF
TIME	T#-106751991167d▼ 7h12m55s807ms	-9.22337204e+18	T#106751991167d▼ 7h12m55s807ms	9.22337204e+18

5.17 ATOLREAL (Any to Long REAL)

5.17.1 Inputs

Input	Data type	Description				
IN	ANY ELEMENTARY	<p>Input IN</p> <p>Range of values: For the data types TIME, ULINT, LINT and LWORD, the modulus of IN should not exceed 2^{53} to prevent rounding errors due to the limited length of the mantissa.</p>				
EN	BOOL	<p>EN is hidden by default. If EN is connected, it is always TRUE. Show EN/ENO: Right-click a function block or function, and select Show EN/ENO from the context menu.</p> <p>As soon as EN or ENO is connected, it can no longer be hidden.</p> <table border="1"> <tr> <td>TRUE</td><td> <p>The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle.</p> <p>ENO Output: The output is TRUE and can be set to FALSE separately based on the internal error detection.</p> </td></tr> <tr> <td>FALSE</td><td> <p>The function block or function is not processed by the user program. The following variables and value field assignment are not assigned.</p> <p>The outputs of a function block display the results of the user program cycle that was lastly processed.</p> <p>The outputs of a function display the default values defined for the data type.</p> <p>As long as the function block or function has not been completely processed, the outputs retain the default values defined for the data type.</p> <p>ENO Output: The output is also set to FALSE.</p> </td></tr> </table>	TRUE	<p>The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle.</p> <p>ENO Output: The output is TRUE and can be set to FALSE separately based on the internal error detection.</p>	FALSE	<p>The function block or function is not processed by the user program. The following variables and value field assignment are not assigned.</p> <p>The outputs of a function block display the results of the user program cycle that was lastly processed.</p> <p>The outputs of a function display the default values defined for the data type.</p> <p>As long as the function block or function has not been completely processed, the outputs retain the default values defined for the data type.</p> <p>ENO Output: The output is also set to FALSE.</p>
TRUE	<p>The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle.</p> <p>ENO Output: The output is TRUE and can be set to FALSE separately based on the internal error detection.</p>					
FALSE	<p>The function block or function is not processed by the user program. The following variables and value field assignment are not assigned.</p> <p>The outputs of a function block display the results of the user program cycle that was lastly processed.</p> <p>The outputs of a function display the default values defined for the data type.</p> <p>As long as the function block or function has not been completely processed, the outputs retain the default values defined for the data type.</p> <p>ENO Output: The output is also set to FALSE.</p>					

5.17.2 Outputs

Output	Data type	Description	
OUT	LREAL	Converts the data type at IN to LREAL. The data type at IN is determined by creating a typecasting line. Also observe the table <i>Converting the Ranges of Values</i> .	
<u>ENO</u>	BOOL	TRUE	If EN = TRUE and the function was processed without errors.
		FALSE	<ul style="list-style-type: none"> • If also EN = FALSE or • Error: IN is out of the range of values for the ANY_REAL data type (+INF, -INF or NaN (not a number)).

5.17.3 Converting the Ranges of Values

Data type at IN	Input lower limit	Output lower limit	Input upper limit	Output upper limit
BOOL	0	0.0	1	1.0
BYTE	16#00	0.0	16#FF	255.0
USINT	0	0.0	255	255.0
UINT	0	0.0	65535	65535.0
UDINT	0	0.0	$2^{32} - 1$	4294967295.0
ULINT	0	0.0	$2^{64} - 1$	1.84467441e+19
INT	-32768	-32768.0	32767	32767.0
SINT	-128	-128.0	127	127.0
DINT	-2^{31}	-2147483648.0	$2^{31} - 1$	2147483647.0
LINT	-2^{63}	$-9.22337204e+18$	$2^{63} - 1$	$9.22337204e+18$
WORD	16#0000	0.0	16#FFFF	65535.0
DWORD	16#0000 0000	0.0	16#FFFF FFFF	4294967295.0
LWORD	16#0000 0000▼ 0000 0000	0.0	16#7FFF FFFF▼ FFFF FFFF	$9.22337204e+18$
REAL	$-3.402823e+38$	$-3.40282306e+38$	$3.402823e+38$	$3.40282306e+38$
LREAL	-1.79769313486▼ 23158e+308	-1.79769313486▼ 232e+308	1.79769313486▼ 23158e+308	1.79769313486▼ 232e+308
TIME	T#-89468321d▼ 19h42m34s400ms	-7730063005354.4	T#89468321d▼ 19h42m34s400ms	7730063005354.4

5.18 TRUNC

This function converts a REAL number into an integer by truncating the decimal places.

5.18.1 Inputs

Input	Data type	Description		
IN	ANY_REAL	Input IN		
EN	BOOL	<p>EN is hidden by default. If EN is connected, it is always TRUE. Show EN/ENO: Right-click a function block or function, and select Show EN/ENO from the context menu.</p> <p>As soon as EN or ENO is connected, it can no longer be hidden.</p>		
		<table border="1"> <tr> <td>TRUE</td><td>The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle. ENO Output: The output is TRUE and can be set to FALSE separately based on the internal error detection.</td></tr> <tr> <td>FALSE</td><td> <p>The function block or function is not processed by the user program. The following variables and value field assignment are not assigned.</p> <p>The outputs of a function block display the results of the user program cycle that was lastly processed.</p> <p>The outputs of a function display the default values defined for the data type.</p> <p>As long as the function block or function has not been completely processed, the outputs retain the default values defined for the data type.</p> <p>ENO Output: The output is also set to FALSE.</p> </td></tr> </table>	TRUE	The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle. ENO Output: The output is TRUE and can be set to FALSE separately based on the internal error detection.
TRUE	The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle. ENO Output: The output is TRUE and can be set to FALSE separately based on the internal error detection.			
FALSE	<p>The function block or function is not processed by the user program. The following variables and value field assignment are not assigned.</p> <p>The outputs of a function block display the results of the user program cycle that was lastly processed.</p> <p>The outputs of a function display the default values defined for the data type.</p> <p>As long as the function block or function has not been completely processed, the outputs retain the default values defined for the data type.</p> <p>ENO Output: The output is also set to FALSE.</p>			

5.18.2 Outputs

Output	Data type	Description																	
OUT	ANY_INT	The decimal places of IN are truncated and the integer part is converted to the defined INT format (→ Conversion Rules).																	
ENO	BOOL	TRUE	If EN = TRUE and the function was processed without errors.																
		FALSE	<ul style="list-style-type: none"> • If also EN = FALSE (then: OUT = 0) or • Error: The result is out of the range of values for OUT. 																
Examples of overflow values: Only the less significant portion that can be represented at OUT is taken into account.		<table border="1"> <thead> <tr> <th>IN</th> <th>Data type</th> <th>TRUNC</th> <th>OUT</th> <th>Data type</th> </tr> </thead> <tbody> <tr> <td>257,234</td> <td>REAL</td> <td>257 = 16#101</td> <td>01</td> <td>USINT</td> </tr> <tr> <td>4863.425364</td> <td>REAL</td> <td>4863 = 16#12FF</td> <td>FF</td> <td>USINT</td> </tr> </tbody> </table>			IN	Data type	TRUNC	OUT	Data type	257,234	REAL	257 = 16#101	01	USINT	4863.425364	REAL	4863 = 16#12FF	FF	USINT
IN	Data type	TRUNC	OUT	Data type															
257,234	REAL	257 = 16#101	01	USINT															
4863.425364	REAL	4863 = 16#12FF	FF	USINT															

5.18.3 Conversion Rules

- When converting from REAL to INT, the value is rounded up to the next integer, if the decimal part is greater than or equal to 0.5. Otherwise, it is rounded down.
- If TRUNC is used, the decimal places of a REAL value are truncated to form the integer value.
- While converting a signed data type to an unsigned data type, the bit pattern of the value is interpreted in the corresponding data format.

Examples:

Value	Converted
-1 (Signed value)	Fhex, or FFFFhex, or FFFFFFFFhex (Unsigned value, depending on the data type)
128 (= 80hex) as USINT	-127 (= 80hex) as SINT
255 (= FFhex) as USINT	-1 (= FFhex) as SINT

5.19 PACK

This function is used to pack up to seven values (of BYTE data type), beginning with the least significant byte.

5.19.1 Inputs

Input	Data type	Description				
IN0	BYTE	Input IN0				
IN1	BYTE	Input IN1				
Extendible	BYTE	to IN7				
<u>EN</u>	BOOL	<p>EN is hidden by default. If EN is connected, it is always TRUE. Show EN/ENO: Right-click a function block or function, and select Show EN/ENO from the context menu.</p> <p>As soon as EN or ENO is connected, it can no longer be hidden.</p> <table border="1"> <tr> <td>TRUE</td><td>The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle. ENO Output: The output is TRUE and can be set to FALSE separately based on the internal error detection.</td></tr> <tr> <td>FALSE</td><td>The function block or function is not processed by the user program. The following variables and value field assignment are not assigned. The outputs of a function block display the results of the user program cycle that was lastly processed. The outputs of a function display the default values defined for the data type. As long as the function block or function has not been completely processed, the outputs retain the default values defined for the data type. ENO Output: The output is also set to FALSE.</td></tr> </table>	TRUE	The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle. ENO Output: The output is TRUE and can be set to FALSE separately based on the internal error detection.	FALSE	The function block or function is not processed by the user program. The following variables and value field assignment are not assigned. The outputs of a function block display the results of the user program cycle that was lastly processed. The outputs of a function display the default values defined for the data type. As long as the function block or function has not been completely processed, the outputs retain the default values defined for the data type. ENO Output: The output is also set to FALSE.
TRUE	The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle. ENO Output: The output is TRUE and can be set to FALSE separately based on the internal error detection.					
FALSE	The function block or function is not processed by the user program. The following variables and value field assignment are not assigned. The outputs of a function block display the results of the user program cycle that was lastly processed. The outputs of a function display the default values defined for the data type. As long as the function block or function has not been completely processed, the outputs retain the default values defined for the data type. ENO Output: The output is also set to FALSE.					

5.19.2 Outputs

Output	Data type	Description											
		<p>The values of IN0...IN7 are output to OUT, beginning with IN0 (the least significant byte). This corresponds to the big endian format.</p> <p>If the data type of OUT receives more bytes than the number occupied by INn, the bytes of OUT that are not occupied are set to 0.</p>											
		Example DWORD: <table border="1"> <thead> <tr> <th>Input</th><th>Type</th><th>Output</th><th>Type</th></tr> </thead> <tbody> <tr> <td>IN0 = 16#01 IN1 = 16#02</td><td>BYTE</td><td>OUT = 16#0201 (More precisely: 16#00000201)</td><td>DWORD (= 4 bytes)</td></tr> </tbody> </table>				Input	Type	Output	Type	IN0 = 16#01 IN1 = 16#02	BYTE	OUT = 16#0201 (More precisely: 16#00000201)	DWORD (= 4 bytes)
Input	Type	Output	Type										
IN0 = 16#01 IN1 = 16#02	BYTE	OUT = 16#0201 (More precisely: 16#00000201)	DWORD (= 4 bytes)										
OUT	ANY_ELEMENTARY	<p>If the data type of OUT receives less bytes than the number occupied by INn (beginning with IN0), only the input values matching OUT are accepted.</p>											
		Example WORD: <table border="1"> <thead> <tr> <th>Input</th><th>Type</th><th>Output</th><th>Type</th></tr> </thead> <tbody> <tr> <td>IN0 = 16#01 IN1 = 16#02 IN2 = 16#03 IN3 = 16#04</td><td>BYTE</td><td>OUT = 16#0201</td><td>WORD (= 2 bytes)</td></tr> </tbody> </table>				Input	Type	Output	Type	IN0 = 16#01 IN1 = 16#02 IN2 = 16#03 IN3 = 16#04	BYTE	OUT = 16#0201	WORD (= 2 bytes)
Input	Type	Output	Type										
IN0 = 16#01 IN1 = 16#02 IN2 = 16#03 IN3 = 16#04	BYTE	OUT = 16#0201	WORD (= 2 bytes)										
ENO	BOOL	<table border="1"> <tr> <td>TRUE</td><td> <ul style="list-style-type: none"> For all data types, except for ANY_REAL: No error can occur, ENO is always TRUE. For REAL data type: The operation was performed without errors. </td></tr> <tr> <td>FALSE</td><td> <ul style="list-style-type: none"> For the REAL data type only: Error: No valid REAL value could be determined. OUT contains one of the following values: +INF (+ infinite), -INF (- infinite), NaN (not a number) </td></tr> </table>	TRUE	<ul style="list-style-type: none"> For all data types, except for ANY_REAL: No error can occur, ENO is always TRUE. For REAL data type: The operation was performed without errors. 	FALSE	<ul style="list-style-type: none"> For the REAL data type only: Error: No valid REAL value could be determined. OUT contains one of the following values: +INF (+ infinite), -INF (- infinite), NaN (not a number) 							
TRUE	<ul style="list-style-type: none"> For all data types, except for ANY_REAL: No error can occur, ENO is always TRUE. For REAL data type: The operation was performed without errors. 												
FALSE	<ul style="list-style-type: none"> For the REAL data type only: Error: No valid REAL value could be determined. OUT contains one of the following values: +INF (+ infinite), -INF (- infinite), NaN (not a number) 												

5.20 UNPACK2

This function is used to transfer the value at IN byte-by-byte to the outputs OUT_0 and OUT_1, starting with the least significant byte.

5.20.1 Inputs

Input	Data type	Description				
IN	ANY_ELEMENTARY	<p>Input IN. IN must not be connected to a constant. Use a variable with the CONST attribute and an initial value instead.</p>				
EN	BOOL	<p>EN is hidden by default. If EN is connected, it is always TRUE. Show EN/ENO: Right-click a function block or function, and select Show EN/ENO from the context menu. As soon as EN or ENO is connected, it can no longer be hidden.</p> <table border="1"> <tr> <td>TRUE</td><td> <p>The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle. ENO Output: The output is TRUE and can be set to FALSE separately based on the internal error detection.</p> </td></tr> <tr> <td>FALSE</td><td> <p>The function block or function is not processed by the user program. The following variables and value field assignment are not assigned. The outputs of a function block display the results of the user program cycle that was lastly processed. The outputs of a function display the default values defined for the data type. As long as the function block or function has not been completely processed, the outputs retain the default values defined for the data type. ENO Output: The output is also set to FALSE.</p> </td></tr> </table>	TRUE	<p>The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle. ENO Output: The output is TRUE and can be set to FALSE separately based on the internal error detection.</p>	FALSE	<p>The function block or function is not processed by the user program. The following variables and value field assignment are not assigned. The outputs of a function block display the results of the user program cycle that was lastly processed. The outputs of a function display the default values defined for the data type. As long as the function block or function has not been completely processed, the outputs retain the default values defined for the data type. ENO Output: The output is also set to FALSE.</p>
TRUE	<p>The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle. ENO Output: The output is TRUE and can be set to FALSE separately based on the internal error detection.</p>					
FALSE	<p>The function block or function is not processed by the user program. The following variables and value field assignment are not assigned. The outputs of a function block display the results of the user program cycle that was lastly processed. The outputs of a function display the default values defined for the data type. As long as the function block or function has not been completely processed, the outputs retain the default values defined for the data type. ENO Output: The output is also set to FALSE.</p>					

5.20.2 Outputs

Output	Data type	Description											
OUT_0, OUT_1	BYTE	<p>Starting with the least significant byte, the value of IN is transferred byte-by-byte to the outputs OUT_0 and OUT_1.</p> <p>OUT_0 = IN(Bit0 to Bit7), or 0x00. OUT_1 = IN(Bit8 to Bit15), or 0x00.</p> <p>If IN contains more than two bytes, only the two less significant bytes are transferred.</p>											
		<table border="1"> <thead> <tr> <th>Input</th><th>Type</th><th>Output</th><th>Type</th></tr> </thead> <tbody> <tr> <td>IN = 16#04030201</td><td>DWORD (= 4 bytes)</td><td>OUT0 = 16#01 OUT1 = 16#02</td><td>BYTE</td></tr> </tbody> </table> <p>If IN contains less than two bytes, the most significant output OUT_1 = 0x00.</p>				Input	Type	Output	Type	IN = 16#04030201	DWORD (= 4 bytes)	OUT0 = 16#01 OUT1 = 16#02	BYTE
Input	Type	Output	Type										
IN = 16#04030201	DWORD (= 4 bytes)	OUT0 = 16#01 OUT1 = 16#02	BYTE										
		<table border="1"> <thead> <tr> <th>Input</th><th>Type</th><th>Output</th><th>Type</th></tr> </thead> <tbody> <tr> <td>IN = 16#01</td><td>BYTE</td><td>OUT0 = 16#01 OUT1 = 16#00</td><td>BYTE</td></tr> </tbody> </table>				Input	Type	Output	Type	IN = 16#01	BYTE	OUT0 = 16#01 OUT1 = 16#00	BYTE
Input	Type	Output	Type										
IN = 16#01	BYTE	OUT0 = 16#01 OUT1 = 16#00	BYTE										
<u>ENO</u>	BOOL	TRUE	<p>If EN is TRUE and the current cycle of the function block or function was processed without errors, ENO is also TRUE.</p> <p>The function cannot result in an error.</p>										
		FALSE	<p>Only if EN is also FALSE or while the function block or function is being processed.</p>										

5.21 UNPACK4

This function is used to transfer the value at IN byte-by-byte to the outputs OUT_0 to OUT_3, starting with the least significant byte.

5.21.1 Inputs

Input	Data type	Description				
IN	ANY_ELEMENTARY	<p>Input IN. IN must not be connected to a constant. Use a variable with the CONST attribute and an initial value instead.</p>				
EN	BOOL	<p>EN is hidden by default. If EN is connected, it is always TRUE. Show EN/ENO: Right-click a function block or function, and select Show EN/ENO from the context menu. As soon as EN or ENO is connected, it can no longer be hidden.</p> <table border="1"> <tr> <td>TRUE</td><td> <p>The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle. ENO Output: The output is TRUE and can be set to FALSE separately based on the internal error detection.</p> </td></tr> <tr> <td>FALSE</td><td> <p>The function block or function is not processed by the user program. The following variables and value field assignment are not assigned. The outputs of a function block display the results of the user program cycle that was lastly processed. The outputs of a function display the default values defined for the data type. As long as the function block or function has not been completely processed, the outputs retain the default values defined for the data type. ENO Output: The output is also set to FALSE.</p> </td></tr> </table>	TRUE	<p>The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle. ENO Output: The output is TRUE and can be set to FALSE separately based on the internal error detection.</p>	FALSE	<p>The function block or function is not processed by the user program. The following variables and value field assignment are not assigned. The outputs of a function block display the results of the user program cycle that was lastly processed. The outputs of a function display the default values defined for the data type. As long as the function block or function has not been completely processed, the outputs retain the default values defined for the data type. ENO Output: The output is also set to FALSE.</p>
TRUE	<p>The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle. ENO Output: The output is TRUE and can be set to FALSE separately based on the internal error detection.</p>					
FALSE	<p>The function block or function is not processed by the user program. The following variables and value field assignment are not assigned. The outputs of a function block display the results of the user program cycle that was lastly processed. The outputs of a function display the default values defined for the data type. As long as the function block or function has not been completely processed, the outputs retain the default values defined for the data type. ENO Output: The output is also set to FALSE.</p>					

5.21.2 Outputs

Output	Data type	Description	
OUT_0, ..., OUT_3	BYTE	<p>Starting with the least significant byte, the value of IN is transferred byte-by-byte to the outputs OUT_0 to OUT_3.</p> <p>OUT_0 = IN(Bit0 bis Bit7), or 00 OUT_1 = IN(Bit8 to Bit15), or 00 OUT_2 = IN(Bit16 to Bit23), or 00 OUT_3 = IN(Bit24 to Bit31), or 00</p> <p>If IN contains more than four bytes, only the four less significant bytes are transferred.</p> <p>If IN contains less than four bytes, the unused most significant output bytes is set to 0x00.</p>	
<u>ENO</u>	BOOL	TRUE	If EN is TRUE and the current cycle of the function block or function was processed without errors, ENO is also TRUE. The function cannot result in an error.
		FALSE	Only if EN is also FALSE or while the function block or function is being processed.

5.22 UNPACK8

This function is used to transfer the value at IN byte-by-byte to the outputs OUT_0 to OUT_7, starting with the least significant byte.

5.22.1 Inputs

Input	Data type	Description				
IN	ANY_ELEMENTARY	<p>Input IN. IN must not be connected to a constant. Use a variable with the CONST attribute and an initial value instead.</p>				
EN	BOOL	<p>EN is hidden by default. If EN is connected, it is always TRUE. Show EN/ENO: Right-click a function block or function, and select Show EN/ENO from the context menu. As soon as EN or ENO is connected, it can no longer be hidden.</p> <table border="1"> <tr> <td>TRUE</td><td> <p>The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle. ENO Output: The output is TRUE and can be set to FALSE separately based on the internal error detection.</p> </td></tr> <tr> <td>FALSE</td><td> <p>The function block or function is not processed by the user program. The following variables and value field assignment are not assigned. The outputs of a function block display the results of the user program cycle that was lastly processed. The outputs of a function display the default values defined for the data type. As long as the function block or function has not been completely processed, the outputs retain the default values defined for the data type. ENO Output: The output is also set to FALSE.</p> </td></tr> </table>	TRUE	<p>The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle. ENO Output: The output is TRUE and can be set to FALSE separately based on the internal error detection.</p>	FALSE	<p>The function block or function is not processed by the user program. The following variables and value field assignment are not assigned. The outputs of a function block display the results of the user program cycle that was lastly processed. The outputs of a function display the default values defined for the data type. As long as the function block or function has not been completely processed, the outputs retain the default values defined for the data type. ENO Output: The output is also set to FALSE.</p>
TRUE	<p>The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle. ENO Output: The output is TRUE and can be set to FALSE separately based on the internal error detection.</p>					
FALSE	<p>The function block or function is not processed by the user program. The following variables and value field assignment are not assigned. The outputs of a function block display the results of the user program cycle that was lastly processed. The outputs of a function display the default values defined for the data type. As long as the function block or function has not been completely processed, the outputs retain the default values defined for the data type. ENO Output: The output is also set to FALSE.</p>					

5.22.2 Outputs

Output	Data type	Description				
OUT_0, ..., OUT_7	BYTE	<p>Starting with the least significant byte, the value of IN is transferred byte-by-byte to the outputs OUT_0 to OUT_7.</p> <p>OUT_0 = IN(Bit0 bis Bit7), or 00 OUT_1 = IN(Bit8 to Bit15), or 00 OUT_2 = IN(Bit16 to Bit23), or 00 OUT_3 = IN(Bit24 to Bit31), or 00 OUT_4 = IN(Bit32 to Bit39), or 00 OUT_5 = IN(Bit40 to Bit47), or 00 OUT_6 = IN(Bit48 to Bit55), or 00 OUT_7 = IN(Bit56 to Bit63), or 00</p> <p>If IN contains less than eight bytes, the unused most significant output bytes is set to 0x00.</p>				
<u>ENO</u>	BOOL	<table border="1"> <tr> <td>TRUE</td><td>If EN is TRUE and the current cycle of the function block or function was processed without errors, ENO is also TRUE. The function cannot result in an error.</td></tr> <tr> <td>FALSE</td><td>Only if EN is also FALSE or while the function block or function is being processed.</td></tr> </table>	TRUE	If EN is TRUE and the current cycle of the function block or function was processed without errors, ENO is also TRUE. The function cannot result in an error.	FALSE	Only if EN is also FALSE or while the function block or function is being processed.
TRUE	If EN is TRUE and the current cycle of the function block or function was processed without errors, ENO is also TRUE. The function cannot result in an error.					
FALSE	Only if EN is also FALSE or while the function block or function is being processed.					

5.23 BIT_PACK

This function is used to pack bit by bit up to eight Boolean values to one data word of type BYTE.

5.23.1 Inputs

Input	Data type	Description				
IN_0, ..., IN_7	BOOL	<p>Inputs of the function.</p> <p>The values of the inputs IN_0 through IN_7 are used in the bits 0 through 7 of the OUT output. IN_0 is the least significant bit (LSB) of OUT, IN_7 is the most significant bit (MSB).</p> <p>The value of unused inputs is 0 (FALSE).</p>				
EN	BOOL	<p>EN is hidden by default. If EN is connected, it is always TRUE.</p> <p>Show EN/ENO: Right-click a function block or function, and select Show EN/ENO from the context menu.</p> <p>As soon as EN or ENO is connected, it can no longer be hidden.</p> <table border="1"> <tr> <td>TRUE</td><td> <p>The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle.</p> <p>ENO output: The ENO output is permanently TRUE. No error can occur.</p> </td></tr> <tr> <td>FALSE</td><td> <p>The function block or function is not processed by the user program. The following variables and value field assignment are not assigned.</p> <p>The outputs of a function block display the results of the user program cycle that was lastly processed.</p> <p>The outputs of a function display the default values defined for the data type.</p> <p>As long as the function block or function has not been completely processed, the outputs retain the default values defined for the data type.</p> <p>ENO Output: The output is also set to FALSE.</p> </td></tr> </table>	TRUE	<p>The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle.</p> <p>ENO output: The ENO output is permanently TRUE. No error can occur.</p>	FALSE	<p>The function block or function is not processed by the user program. The following variables and value field assignment are not assigned.</p> <p>The outputs of a function block display the results of the user program cycle that was lastly processed.</p> <p>The outputs of a function display the default values defined for the data type.</p> <p>As long as the function block or function has not been completely processed, the outputs retain the default values defined for the data type.</p> <p>ENO Output: The output is also set to FALSE.</p>
TRUE	<p>The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle.</p> <p>ENO output: The ENO output is permanently TRUE. No error can occur.</p>					
FALSE	<p>The function block or function is not processed by the user program. The following variables and value field assignment are not assigned.</p> <p>The outputs of a function block display the results of the user program cycle that was lastly processed.</p> <p>The outputs of a function display the default values defined for the data type.</p> <p>As long as the function block or function has not been completely processed, the outputs retain the default values defined for the data type.</p> <p>ENO Output: The output is also set to FALSE.</p>					

5.23.2 Outputs

Output	Data type	Description								
OUT	BYTE	Gepackte Bits von IN_0 bis IN_7.								
		Beispiele:								
		IN_7	IN_6	IN_5	IN_4	IN_3	IN_2	IN_1	IN_0	OUT
		0	0	0	0	0	0	0	1	0x1
		1	0	0	0	0	0	0	1	0x81
		1	1	1	1	1	1	1	1	0xFF
<u>ENO</u>		TRUE	If EN is TRUE and the current cycle of the function block or function was processed without errors, ENO is also TRUE. The function cannot result in an error.							
		FALSE	Only if EN is also FALSE or while the function block or function is being processed.							

5.24 BIT_UNPACK

This function is used to unpack bit by bit one byte to eight Boolean values.

5.24.1 Inputs

Input	Data type	Description				
IN	BYTE	<p>Input of the function.</p> <p>The value of the IN input is parsed bit by bit and transferred to the outputs OUT_0 through OUT_7. OUT_0 contains the least significant bit (LSB) of IN, OUT_7 the most significant bit (MSB).</p>				
EN	BOOL	<p>EN is hidden by default. If EN is connected, it is always TRUE.</p> <p>Show EN/ENO: Right-click a function block or function, and select Show EN/ENO from the context menu.</p> <p>As soon as EN or ENO is connected, it can no longer be hidden.</p> <table border="1"> <tr> <td>TRUE</td><td> <p>The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle.</p> <p>ENO output: The ENO output is permanently TRUE. No error can occur.</p> </td></tr> <tr> <td>FALSE</td><td> <p>The function block or function is not processed by the user program. The following variables and value field assignment are not assigned.</p> <p>The outputs of a function block display the results of the user program cycle that was lastly processed.</p> <p>The outputs of a function display the default values defined for the data type.</p> <p>As long as the function block or function has not been completely processed, the outputs retain the default values defined for the data type.</p> <p>ENO Output: The output is also set to FALSE.</p> </td></tr> </table>	TRUE	<p>The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle.</p> <p>ENO output: The ENO output is permanently TRUE. No error can occur.</p>	FALSE	<p>The function block or function is not processed by the user program. The following variables and value field assignment are not assigned.</p> <p>The outputs of a function block display the results of the user program cycle that was lastly processed.</p> <p>The outputs of a function display the default values defined for the data type.</p> <p>As long as the function block or function has not been completely processed, the outputs retain the default values defined for the data type.</p> <p>ENO Output: The output is also set to FALSE.</p>
TRUE	<p>The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle.</p> <p>ENO output: The ENO output is permanently TRUE. No error can occur.</p>					
FALSE	<p>The function block or function is not processed by the user program. The following variables and value field assignment are not assigned.</p> <p>The outputs of a function block display the results of the user program cycle that was lastly processed.</p> <p>The outputs of a function display the default values defined for the data type.</p> <p>As long as the function block or function has not been completely processed, the outputs retain the default values defined for the data type.</p> <p>ENO Output: The output is also set to FALSE.</p>					

5.24.2 Outputs

Output	Data type	Description																																												
OUT_0, ..., OUT_7	BOOL	Entpackte Bits von IN. Beispiele:																																												
		<table border="1" style="margin-left: auto; margin-right: auto;"> <thead> <tr> <th>IN</th><th>OUT_7</th><th>OUT_6</th><th>OUT_5</th><th>OUT_4</th><th>OUT_3</th><th>OUT_2</th><th>OUT_1</th><th>OUT_0</th></tr> </thead> <tbody> <tr> <td>0x1</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>1</td></tr> <tr> <td>0x81</td><td>1</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>1</td></tr> <tr> <td>0xFF</td><td>1</td><td>1</td><td>1</td><td>1</td><td>1</td><td>1</td><td>1</td><td>1</td></tr> </tbody> </table>									IN	OUT_7	OUT_6	OUT_5	OUT_4	OUT_3	OUT_2	OUT_1	OUT_0	0x1	0	0	0	0	0	0	0	1	0x81	1	0	0	0	0	0	0	1	0xFF	1	1	1	1	1	1	1	1
IN	OUT_7	OUT_6	OUT_5	OUT_4	OUT_3	OUT_2	OUT_1	OUT_0																																						
0x1	0	0	0	0	0	0	0	1																																						
0x81	1	0	0	0	0	0	0	1																																						
0xFF	1	1	1	1	1	1	1	1																																						
<u>ENO</u>		TRUE	If EN is TRUE and the current cycle of the function block or function was processed without errors, ENO is also TRUE. The function cannot result in an error.																																											
		FALSE	Only if EN is also FALSE or while the function block or function is being processed.																																											

6 COUNTER Sub-Library

6.1 Values Allowed for PV

In all counter function blocks, the default value allowed for PV is identical with the maximum or minimum absolute value of the used data type, except for ANY_REAL.

For ANY_REAL, the following limits apply:

Data type	PV(max)	PV(min)
REAL	6777216.0	-16777216.0
LREAL	9007199254740992.0	-9007199254740992.0

6.2 CTD (Counter Down)

This function block is a down counter with reset and preset options.

6.2.1 Inputs

Input	Data type	Description
>CD	BOOL	Used to decrement the counter value (CV) by 1 when the signal changes from FALSE to TRUE.
LD	BOOL	Used to set the counter value (CV) to the preset value (PV). The counter cannot be decremented as long as LD = TRUE.
PV	ANY_NUM	<p>Counter initial value (preset value).</p> <p>If the counter value is less than or equal to, the Q output is TRUE.</p> <p>Ranges of values for the data types REAL and LREAL:</p> <p>REAL -16777216.0 ... 16777216.0 LREAL -9007199254740992.0 ... 9007199254740992.0</p> <p>For all other data types: The range limits of the selected data types apply.</p> <div style="border: 2px solid red; padding: 5px;">  For the ANY_REAL data type, PV can be loaded with a value that exceeds the absolute value of the ANY_REAL limits, with no error being output to the ENO output. This causes the counter to malfunction! </div>

Input	Data type	Description		
EN	BOOL	<p>EN is hidden by default. If EN is connected, it is always TRUE.</p> <p>Show EN/ENO: Right-click a function block or function, and select Show EN/ENO from the context menu.</p> <p>As soon as EN or ENO is connected, it can no longer be hidden.</p>		
		<table border="1"> <tr> <td>TRUE</td><td>The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle. ENO Output: The output is TRUE and can be set to FALSE separately based on the internal error detection.</td></tr> <tr> <td>FALSE</td><td>The function block or function is not processed by the user program. The following variables and value field assignment are not assigned. The outputs of a function block display the results of the user program cycle that was lastly processed. The outputs of a function display the default values defined for the data type. As long as the function block or function has not been completely processed, the outputs retain the default values defined for the data type. ENO Output: The output is also set to FALSE.</td></tr> </table>	TRUE	The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle. ENO Output: The output is TRUE and can be set to FALSE separately based on the internal error detection.
TRUE	The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle. ENO Output: The output is TRUE and can be set to FALSE separately based on the internal error detection.			
FALSE	The function block or function is not processed by the user program. The following variables and value field assignment are not assigned. The outputs of a function block display the results of the user program cycle that was lastly processed. The outputs of a function display the default values defined for the data type. As long as the function block or function has not been completely processed, the outputs retain the default values defined for the data type. ENO Output: The output is also set to FALSE.			

6.2.2 Outputs

Output	Data type	Description	
Q	BOOL	TRUE	If $CV \leq 0$ Use the inputs PV and LD to set the counter's initial value.
		FALSE	If $CV > 0$.
CV	ANY_NUM	<p>Current counter value.</p> <p>The data type at CV is identical with the data type at PV. The minimum value of CV is defined by the used data type (→ Data Type and Range of Values).</p>	
ENO	BOOL	TRUE	If EN = TRUE and the function was processed without errors.
		TRUE → FALSE → TRUE	If $CV = 0$ and the counter is further decremented, ENO is set to FALSE for the current cycle.
		FALSE	<ul style="list-style-type: none"> If also EN = FALSE or Error: PV exceeds the value for the REAL or LREAL data type.

6.2.3 Implementation in Structured Text (ST)

```

IF LD
  THEN CV := PV;
ELSIF CD AND (CV > PVmin)
  THEN CV := CV-1;
END_IF;
Q := (CV <= 0);

```

6.3 CTU (Counter Up)

This function block is an up counter with reset and preset options.

6.3.1 Inputs

Input	Data type	Description
>CU	BOOL	Used to increment the counter value (CV) by 1 when the signal changes from FALSE to TRUE.
R	BOOL	Used to set the counter value to 0 (CV = 0). The counter cannot be incremented as long as R = TRUE.
PV	ANY_NUM	Defines the maximum value of the counter. If CV ≥ PV, the Q output is TRUE. Ranges of values for the data types REAL and LREAL: REAL: -16777216.0 ... 16777216.0 LREAL: -9007199254740992.0 ... 9007199254740992.0 For all other data types: The range limits of the selected data types apply.

Input	Data type	Description				
EN	BOOL	<p>EN is hidden by default. If EN is connected, it is always TRUE.</p> <p>Show EN/ENO: Right-click a function block or function, and select Show EN/ENO from the context menu.</p> <p>As soon as EN or ENO is connected, it can no longer be hidden.</p> <table border="1"> <tr> <td>TRUE</td><td>The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle. ENO Output: The output is TRUE and can be set to FALSE separately based on the internal error detection.</td></tr> <tr> <td>FALSE</td><td>The function block or function is not processed by the user program. The following variables and value field assignment are not assigned. The outputs of a function block display the results of the user program cycle that was lastly processed. The outputs of a function display the default values defined for the data type. As long as the function block or function has not been completely processed, the outputs retain the default values defined for the data type. ENO Output: The output is also set to FALSE.</td></tr> </table>	TRUE	The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle. ENO Output: The output is TRUE and can be set to FALSE separately based on the internal error detection.	FALSE	The function block or function is not processed by the user program. The following variables and value field assignment are not assigned. The outputs of a function block display the results of the user program cycle that was lastly processed. The outputs of a function display the default values defined for the data type. As long as the function block or function has not been completely processed, the outputs retain the default values defined for the data type. ENO Output: The output is also set to FALSE.
TRUE	The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle. ENO Output: The output is TRUE and can be set to FALSE separately based on the internal error detection.					
FALSE	The function block or function is not processed by the user program. The following variables and value field assignment are not assigned. The outputs of a function block display the results of the user program cycle that was lastly processed. The outputs of a function display the default values defined for the data type. As long as the function block or function has not been completely processed, the outputs retain the default values defined for the data type. ENO Output: The output is also set to FALSE.					

6.3.2 Outputs

Output	Data type	Description	
Q	BOOL	TRUE	If CV \geq PV. Use the input R to reset the counter.
		FALSE	If CV < PV.
CV	ANY_NUM	Current counter value. The data type at CV is identical with the data type at PV. The maximum value of CV is defined by the used data type (→ Data Type and Range of Values).	
ENO	BOOL	TRUE	If EN = TRUE and the function was processed without errors.
		TRUE → FALSE → TRUE	If CV = PV and the counter is further incremented, ENO is set to FALSE for the current cycle.
		FALSE	<ul style="list-style-type: none"> • If also EN = FALSE or • Error: PV exceeds the value for the REAL or LREAL data type.

6.3.3 Implementation in Structured Text (ST)

```

IF R
  THEN CV := 0;
ELSIF CU AND (CV < PVmax)
  THEN CV := CV+1;
END_IF;
Q := (CV >= PV);

```

6.4 CTUD (Counter Up/Down)

This function block is an up and down counter with separate inputs for counting up and counting down.

6.4.1 Inputs

Input	Data type	Description
>CU	BOOL	Used to increment the counter value (CV) by 1 when the signal changes from FALSE to TRUE.
>CD	BOOL	Used to decrement the counter value (CV) by 1 when the signal changes from FALSE to TRUE.
R	BOOL	Used to set the counter value to 0 (CV = 0). As a result, QD is set to TRUE. The counter can be neither incremented nor decremented as long as R = TRUE. If R and LD are TRUE simultaneously, R is dominant.
LD	BOOL	Used to set the counter value (CV) to the preset value (PV). As a result, QU is set to TRUE. The counter can be neither incremented nor decremented as long as LD = TRUE.

Input	Data type	Description		
PV	ANY_NUM	<ul style="list-style-type: none"> For up counters: If $CV \geq PV$, the QU output is TRUE. For down counters: Counter initial value (preset value). If $CV \leq 0$, the QD output is TRUE. <p>Ranges of values for the data types REAL and LREAL:</p> <p>REAL: -16777216.0 ... 16777216.0 LREAL: -9007199254740992.0 ... 9007199254740992.0</p> <p>For all other data types: The range limits of the selected data types apply.</p> <div style="border: 2px solid red; padding: 5px; margin-top: 10px;">  For the ANY_REAL data type, PV can be loaded with a value that exceeds the absolute value of the ANY_REAL limits, with no error being output to the ENO output. This causes the counter to malfunction! </div>		
<u>EN</u>	BOOL	<p>EN is hidden by default. If EN is connected, it is always TRUE. Show EN/ENO: Right-click a function block or function, and select Show EN/ENO from the context menu.</p> <p>As soon as EN or ENO is connected, it can no longer be hidden.</p> <table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td style="padding: 5px; vertical-align: top;"> <p>TRUE</p> <p>The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle.</p> <p>ENO Output: The output is TRUE and can be set to FALSE separately based on the internal error detection.</p> </td> <td style="padding: 5px; vertical-align: top;"> <p>FALSE</p> <p>The function block or function is not processed by the user program. The following variables and value field assignment are not assigned.</p> <p>The outputs of a function block display the results of the user program cycle that was lastly processed.</p> <p>The outputs of a function display the default values defined for the data type.</p> <p>As long as the function block or function has not been completely processed, the outputs retain the default values defined for the data type.</p> <p>ENO Output: The output is also set to FALSE.</p> </td> </tr> </table>	<p>TRUE</p> <p>The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle.</p> <p>ENO Output: The output is TRUE and can be set to FALSE separately based on the internal error detection.</p>	<p>FALSE</p> <p>The function block or function is not processed by the user program. The following variables and value field assignment are not assigned.</p> <p>The outputs of a function block display the results of the user program cycle that was lastly processed.</p> <p>The outputs of a function display the default values defined for the data type.</p> <p>As long as the function block or function has not been completely processed, the outputs retain the default values defined for the data type.</p> <p>ENO Output: The output is also set to FALSE.</p>
<p>TRUE</p> <p>The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle.</p> <p>ENO Output: The output is TRUE and can be set to FALSE separately based on the internal error detection.</p>	<p>FALSE</p> <p>The function block or function is not processed by the user program. The following variables and value field assignment are not assigned.</p> <p>The outputs of a function block display the results of the user program cycle that was lastly processed.</p> <p>The outputs of a function display the default values defined for the data type.</p> <p>As long as the function block or function has not been completely processed, the outputs retain the default values defined for the data type.</p> <p>ENO Output: The output is also set to FALSE.</p>			

6.4.2 Outputs

Output	Data type	Description	
QU	BOOL	TRUE	If $CV \geq PV$. Use the input R to reset the counter.
		FALSE	If $CV < PV$.
QD	BOOL	TRUE	If $CV \leq 0$ Use the inputs PV and LD to set the counter's initial value.
		FALSE	If $CV > 0$.
CV	ANY_NUM	Current counter value. The data type at CV is identical with the data type at PV. The range of values for CV is defined by the used data type (→ Data Type and Range of Values).	
ENO	BOOL	TRUE	If EN = TRUE and the function was processed without errors.
		TRUE → FALSE → TRUE	<ul style="list-style-type: none"> For up counters: If $CV = PV$ and the counter is further incremented, ENO is set to FALSE for the current cycle. For down counters: If $CV = 0$ and the counter is further decremented, ENO is set to FALSE for the current cycle.
		FALSE	<ul style="list-style-type: none"> If also EN = FALSE or Error: PV exceeds the value for the REAL or LREAL data type.

6.4.3 Implementation in Structured Text (ST)

```

IF R
  THEN CV := 0;
ELSIF LD
  THEN CV := PV;
ELSE
  IF NOT (CU AND CD) THEN
    IF CU AND (CV < PVmax)
      THEN CV := CV+1;
    ELSIF CD AND (CV > PVmin)
      THEN CV := CV-1;
  END_IF;
  END_IF;
  END_IF;
  QU := (CV >= PV);
  QU := (CV <= 0);

```


7 EDGE Sub-Library

7.1 F_TRIG

This function block is used to detect a falling edge.

7.1.1 Inputs

Input	Data type	Description				
CLK	BOOL	Pulsed input				
EN	BOOL	<p>EN is hidden by default. If EN is connected, it is always TRUE. Show EN/ENO: Right-click a function block or function, and select Show EN/ENO from the context menu.</p> <p>As soon as EN or ENO is connected, it can no longer be hidden.</p> <table border="1"> <tr> <td>TRUE</td> <td> <p>The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle.</p> <p>ENO Output: The output is TRUE and can be set to FALSE separately based on the internal error detection.</p> </td> </tr> <tr> <td>FALSE</td> <td> <p>The function block or function is not processed by the user program. The following variables and value field assignment are not assigned.</p> <p>The outputs of a function block display the results of the user program cycle that was lastly processed.</p> <p>The outputs of a function display the default values defined for the data type.</p> <p>As long as the function block or function has not been completely processed, the outputs retain the default values defined for the data type.</p> <p>ENO Output: The output is also set to FALSE.</p> </td> </tr> </table>	TRUE	<p>The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle.</p> <p>ENO Output: The output is TRUE and can be set to FALSE separately based on the internal error detection.</p>	FALSE	<p>The function block or function is not processed by the user program. The following variables and value field assignment are not assigned.</p> <p>The outputs of a function block display the results of the user program cycle that was lastly processed.</p> <p>The outputs of a function display the default values defined for the data type.</p> <p>As long as the function block or function has not been completely processed, the outputs retain the default values defined for the data type.</p> <p>ENO Output: The output is also set to FALSE.</p>
TRUE	<p>The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle.</p> <p>ENO Output: The output is TRUE and can be set to FALSE separately based on the internal error detection.</p>					
FALSE	<p>The function block or function is not processed by the user program. The following variables and value field assignment are not assigned.</p> <p>The outputs of a function block display the results of the user program cycle that was lastly processed.</p> <p>The outputs of a function display the default values defined for the data type.</p> <p>As long as the function block or function has not been completely processed, the outputs retain the default values defined for the data type.</p> <p>ENO Output: The output is also set to FALSE.</p>					

7.1.2 Outputs

Output	Data type	Description	
Q	BOOL	TRUE	After a falling edge at CLK, Q is set to TRUE for the duration of a cycle.
		FALSE	No falling edge at CLK.
ENO	BOOL	TRUE	If EN is TRUE and the current cycle of the function block or function was processed without errors, ENO is also TRUE. The function cannot result in an error.
		FALSE	Only if EN is also FALSE or while the function block or function is being processed.

7.1.3 Implementation in Structured Text (ST)

```
FUNCTION_BLOCK F_TRIG
VAR_INPUT CLK: BOOL; END_VAR
```

```

VAR_OUTPUT Q: BOOL; END_VAR
VAR M: BOOL := 1; END_VAR
Q := NOT CLK AND NOT M;
M := NOT CLK;
END_FUNCTION_BLOCK

```

7.2 R_TRIG

This function block is used to detect a rising edge.

7.2.1 Inputs

Input	Data type	Description				
CLK	BOOL	Pulsed input				
EN	BOOL	<p>EN is hidden by default. If EN is connected, it is always TRUE.</p> <p>Show EN/ENO: Right-click a function block or function, and select Show EN/ENO from the context menu.</p> <p>As soon as EN or ENO is connected, it can no longer be hidden.</p> <table border="1"> <tr> <td>TRUE</td><td>The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle. ENO Output: The output is TRUE and can be set to FALSE separately based on the internal error detection.</td></tr> <tr> <td>FALSE</td><td>The function block or function is not processed by the user program. The following variables and value field assignment are not assigned. The outputs of a function block display the results of the user program cycle that was lastly processed. The outputs of a function display the default values defined for the data type. As long as the function block or function has not been completely processed, the outputs retain the default values defined for the data type. ENO Output: The output is also set to FALSE.</td></tr> </table>	TRUE	The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle. ENO Output: The output is TRUE and can be set to FALSE separately based on the internal error detection.	FALSE	The function block or function is not processed by the user program. The following variables and value field assignment are not assigned. The outputs of a function block display the results of the user program cycle that was lastly processed. The outputs of a function display the default values defined for the data type. As long as the function block or function has not been completely processed, the outputs retain the default values defined for the data type. ENO Output: The output is also set to FALSE.
TRUE	The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle. ENO Output: The output is TRUE and can be set to FALSE separately based on the internal error detection.					
FALSE	The function block or function is not processed by the user program. The following variables and value field assignment are not assigned. The outputs of a function block display the results of the user program cycle that was lastly processed. The outputs of a function display the default values defined for the data type. As long as the function block or function has not been completely processed, the outputs retain the default values defined for the data type. ENO Output: The output is also set to FALSE.					

7.2.2 Outputs

Output	Data type	Description	
Q	BOOL	TRUE	After a rising edge at CLK, Q is set to TRUE for the duration of a cycle.
		FALSE	No rising edge at CLK.
ENO	BOOL	TRUE	If EN is TRUE and the current cycle of the function block or function was processed without errors, ENO is also TRUE. The function cannot result in an error.
		FALSE	Only if EN is also FALSE or while the function block or function is being processed.

7.2.3 Implementation in Structured Text (ST)

```

FUNCTION_BLOCK R_TRIG
  VAR_INPUT CLK: BOOL; END_VAR
  VAR_OUTPUT Q: BOOL; END_VAR
  VAR M: BOOL := 0; END_VAR
  Q := CLK AND NOT M;
  M := CLK;
END_FUNCTION_BLOCK

```


8 NUMERIC Sub-Library

8.1 ABS (Absolute Value)

The ABS function calculates the modulus (absolute value) of a function value.

8.1.1 Inputs

Input	Data type	Description		
IN	ANY_NUM	<p>Input of the function block.</p> <p>The range of values depends on the signal at IN1.</p>		
<u>EN</u>	BOOL	<p>EN is hidden by default. If EN is connected, it is always TRUE.</p> <p>Show EN/ENO: Right-click a function block or function, and select Show EN/ENO from the context menu.</p> <p>As soon as EN or ENO is connected, it can no longer be hidden.</p>		
		<table border="1"> <tr> <td>TRUE</td><td> <p>The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle.</p> <p>ENO Output: The output is TRUE and can be set to FALSE separately based on the internal error detection.</p> </td></tr> <tr> <td>FALSE</td><td> <p>The function block or function is not processed by the user program. The following variables and value field assignment are not assigned.</p> <p>The outputs of a function block display the results of the user program cycle that was lastly processed.</p> <p>The outputs of a function display the default values defined for the data type.</p> <p>As long as the function block or function has not been completely processed, the outputs retain the default values defined for the data type.</p> <p>ENO Output: The output is also set to FALSE.</p> </td></tr> </table>	TRUE	<p>The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle.</p> <p>ENO Output: The output is TRUE and can be set to FALSE separately based on the internal error detection.</p>
TRUE	<p>The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle.</p> <p>ENO Output: The output is TRUE and can be set to FALSE separately based on the internal error detection.</p>			
FALSE	<p>The function block or function is not processed by the user program. The following variables and value field assignment are not assigned.</p> <p>The outputs of a function block display the results of the user program cycle that was lastly processed.</p> <p>The outputs of a function display the default values defined for the data type.</p> <p>As long as the function block or function has not been completely processed, the outputs retain the default values defined for the data type.</p> <p>ENO Output: The output is also set to FALSE.</p>			

8.1.2 Outputs

Output	Data type	Description				
OUT	ANY_NUM	<ul style="list-style-type: none"> Absolute value of IN, if OUT is within the IN range of values. If IN contains the largest negative value of an integer data type, this value is output unchanged to OUT since the corresponding positive value does not exist. See also ENO! If IN is NaN, OUT is NaN as well. If IN is +INF, OUT is +INF as well. If IN is -INF, OUT is +INF. 				
<u>ENO</u>		<table border="1"> <tr> <td>TRUE</td><td>If EN = TRUE and the function was processed without errors.</td></tr> <tr> <td>FALSE</td><td> <ul style="list-style-type: none"> If also EN = FALSE or Error: IN contains the largest negative value of an integer data type. The result is out of range. Example of INT: ABS(-32768) -> OUT = -32768 </td></tr> </table>	TRUE	If EN = TRUE and the function was processed without errors.	FALSE	<ul style="list-style-type: none"> If also EN = FALSE or Error: IN contains the largest negative value of an integer data type. The result is out of range. Example of INT: ABS(-32768) -> OUT = -32768
TRUE	If EN = TRUE and the function was processed without errors.					
FALSE	<ul style="list-style-type: none"> If also EN = FALSE or Error: IN contains the largest negative value of an integer data type. The result is out of range. Example of INT: ABS(-32768) -> OUT = -32768 					

8.2 SQRT (Square Root)

The SQTR function calculates the square root of the function value. For negative input values, OUT = NaN.

8.2.1 Inputs

Input	Data type	Description				
IN	ANY_REAL	<p>Input IN</p> <p>Range of values: IN ≥ 0.0</p>				
<u>EN</u>	BOOL	<p>EN is hidden by default. If EN is connected, it is always TRUE.</p> <p>Show EN/ENO: Right-click a function block or function, and select Show EN/ENO from the context menu.</p> <p>As soon as EN or ENO is connected, it can no longer be hidden.</p> <table border="1"> <tr> <td>TRUE</td><td> <p>The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle.</p> <p>ENO Output: The output is TRUE and can be set to FALSE separately based on the internal error detection.</p> </td></tr> <tr> <td>FALSE</td><td> <p>The function block or function is not processed by the user program. The following variables and value field assignment are not assigned.</p> <p>The outputs of a function block display the results of the user program cycle that was lastly processed.</p> <p>The outputs of a function display the default values defined for the data type.</p> <p>As long as the function block or function has not been completely processed, the outputs retain the default values defined for the data type.</p> <p>ENO Output: The output is also set to FALSE.</p> </td></tr> </table>	TRUE	<p>The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle.</p> <p>ENO Output: The output is TRUE and can be set to FALSE separately based on the internal error detection.</p>	FALSE	<p>The function block or function is not processed by the user program. The following variables and value field assignment are not assigned.</p> <p>The outputs of a function block display the results of the user program cycle that was lastly processed.</p> <p>The outputs of a function display the default values defined for the data type.</p> <p>As long as the function block or function has not been completely processed, the outputs retain the default values defined for the data type.</p> <p>ENO Output: The output is also set to FALSE.</p>
TRUE	<p>The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle.</p> <p>ENO Output: The output is TRUE and can be set to FALSE separately based on the internal error detection.</p>					
FALSE	<p>The function block or function is not processed by the user program. The following variables and value field assignment are not assigned.</p> <p>The outputs of a function block display the results of the user program cycle that was lastly processed.</p> <p>The outputs of a function display the default values defined for the data type.</p> <p>As long as the function block or function has not been completely processed, the outputs retain the default values defined for the data type.</p> <p>ENO Output: The output is also set to FALSE.</p>					

8.2.2 Outputs

Output	Data type	Description	
OUT	ANY_REAL	<p>Square root of IN.</p> <ul style="list-style-type: none">• If IN is NaN, OUT is NaN as well.• If IN is +INF, OUT is +INF as well.• If IN is -INF, OUT is NaN.	
<u>ENO</u>	BOOL	TRUE	The mathematical function was performed without errors.
		FALSE	Error: IN is out of range (IN is negative). The value of OUT is NaN (not a number).

8.3 LN (Natural Logarithm)

The LN function calculates the natural logarithm to base e (\ln) of input values > 0.0 , where e is the Euler's number (2.7182818284590...).

8.3.1 Inputs

Input	Data type	Description				
IN	ANY_REAL	<p>Input IN</p> <p>Range of values: IN > 0.0</p>				
<u>EN</u>	BOOL	<p>EN is hidden by default. If EN is connected, it is always TRUE.</p> <p>Show EN/ENO: Right-click a function block or function, and select Show EN/ENO from the context menu.</p> <p>As soon as EN or ENO is connected, it can no longer be hidden.</p> <table border="1"> <tr> <td>TRUE</td><td> <p>The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle.</p> <p>ENO Output: The output is TRUE and can be set to FALSE separately based on the internal error detection.</p> </td></tr> <tr> <td>FALSE</td><td> <p>The function block or function is not processed by the user program. The following variables and value field assignment are not assigned.</p> <p>The outputs of a function block display the results of the user program cycle that was lastly processed.</p> <p>The outputs of a function display the default values defined for the data type.</p> <p>As long as the function block or function has not been completely processed, the outputs retain the default values defined for the data type.</p> <p>ENO Output: The output is also set to FALSE.</p> </td></tr> </table>	TRUE	<p>The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle.</p> <p>ENO Output: The output is TRUE and can be set to FALSE separately based on the internal error detection.</p>	FALSE	<p>The function block or function is not processed by the user program. The following variables and value field assignment are not assigned.</p> <p>The outputs of a function block display the results of the user program cycle that was lastly processed.</p> <p>The outputs of a function display the default values defined for the data type.</p> <p>As long as the function block or function has not been completely processed, the outputs retain the default values defined for the data type.</p> <p>ENO Output: The output is also set to FALSE.</p>
TRUE	<p>The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle.</p> <p>ENO Output: The output is TRUE and can be set to FALSE separately based on the internal error detection.</p>					
FALSE	<p>The function block or function is not processed by the user program. The following variables and value field assignment are not assigned.</p> <p>The outputs of a function block display the results of the user program cycle that was lastly processed.</p> <p>The outputs of a function display the default values defined for the data type.</p> <p>As long as the function block or function has not been completely processed, the outputs retain the default values defined for the data type.</p> <p>ENO Output: The output is also set to FALSE.</p>					

8.3.2 Outputs

Output	Data type	Description	
OUT	BOOL	Natural logarithm of IN. <ul style="list-style-type: none"> • If IN is less than 0.0, OUT is NaN. • If IN is 0.0, OUT is -INF. • If IN is NaN, OUT is NaN as well. • If IN is +INF, OUT is +INF as well. • If IN is -INF, OUT is NaN. 	
<u>ENO</u>	BOOL	TRUE	<ul style="list-style-type: none"> • EN = TRUE and • The mathematical function was performed without errors.
		FALSE	<ul style="list-style-type: none"> • EN = FALSE or • Error: The lower range limit of IN was violated. The mathematical function could not be performed. The value of OUT is NaN (not a number).

8.4 LOG (Base-10 Logarithm)

The LOG function calculates the logarithm to base 10 (\lg) for input values > 0.0 .

8.4.1 Inputs

Input	Data type	Description				
IN	ANY_REAL	<p>Input IN</p> <p>Range of values: IN > 0.0</p>				
<u>EN</u>	BOOL	<p>EN is hidden by default. If EN is connected, it is always TRUE.</p> <p>Show EN/ENO: Right-click a function block or function, and select Show EN/ENO from the context menu.</p> <p>As soon as EN or ENO is connected, it can no longer be hidden.</p> <table border="1"> <tr> <td>TRUE</td><td> <p>The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle.</p> <p>ENO Output: The output is TRUE and can be set to FALSE separately based on the internal error detection.</p> </td></tr> <tr> <td>FALSE</td><td> <p>The function block or function is not processed by the user program. The following variables and value field assignment are not assigned.</p> <p>The outputs of a function block display the results of the user program cycle that was lastly processed.</p> <p>The outputs of a function display the default values defined for the data type.</p> <p>As long as the function block or function has not been completely processed, the outputs retain the default values defined for the data type.</p> <p>ENO Output: The output is also set to FALSE.</p> </td></tr> </table>	TRUE	<p>The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle.</p> <p>ENO Output: The output is TRUE and can be set to FALSE separately based on the internal error detection.</p>	FALSE	<p>The function block or function is not processed by the user program. The following variables and value field assignment are not assigned.</p> <p>The outputs of a function block display the results of the user program cycle that was lastly processed.</p> <p>The outputs of a function display the default values defined for the data type.</p> <p>As long as the function block or function has not been completely processed, the outputs retain the default values defined for the data type.</p> <p>ENO Output: The output is also set to FALSE.</p>
TRUE	<p>The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle.</p> <p>ENO Output: The output is TRUE and can be set to FALSE separately based on the internal error detection.</p>					
FALSE	<p>The function block or function is not processed by the user program. The following variables and value field assignment are not assigned.</p> <p>The outputs of a function block display the results of the user program cycle that was lastly processed.</p> <p>The outputs of a function display the default values defined for the data type.</p> <p>As long as the function block or function has not been completely processed, the outputs retain the default values defined for the data type.</p> <p>ENO Output: The output is also set to FALSE.</p>					

8.4.2 Outputs

Output	Data type	Description	
OUT	BOOL	Base-10 logarithm of IN. <ul style="list-style-type: none"> If IN is less than 0.0, OUT is NaN. If IN is 0.0, OUT is -INF. If IN is NaN, OUT is NaN as well. If IN is +INF, OUT is +INF as well. If IN is -INF, OUT is NaN. 	
<u>ENO</u>	BOOL	TRUE	<ul style="list-style-type: none"> EN = TRUE and The mathematical function was performed without errors.
		FALSE	<ul style="list-style-type: none"> EN = FALSE or Error: The lower range limit of IN was violated. The mathematical function could not be performed. The value of OUT is NaN (not a number).

8.5 EXP (Exponential Function e^x)

The EXP function calculates the e^x of the function value. The Euler's number e is raised to the power of the function value x.

8.5.1 Inputs

Input	Data type	Description				
IN	ANY_REAL	<p>Input IN The values must be within the following limits: REAL: $-87.336544751252633835568719103432 \leq IN \leq 88.722839053130621324601674778549$ LREAL: $-708.39641853226410621681158499121 \leq IN \leq 709.7827128933839967276924307167$</p>				
EN	BOOL	<p>EN is hidden by default. If EN is connected, it is always TRUE. Show EN/ENO: Right-click a function block or function, and select Show EN/ENO from the context menu.</p> <p>As soon as EN or ENO is connected, it can no longer be hidden.</p> <table border="1"> <tr> <td>TRUE</td><td> <p>The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle.</p> <p>ENO Output: The output is TRUE and can be set to FALSE separately based on the internal error detection.</p> </td></tr> <tr> <td>FALSE</td><td> <p>The function block or function is not processed by the user program. The following variables and value field assignment are not assigned.</p> <p>The outputs of a function block display the results of the user program cycle that was lastly processed.</p> <p>The outputs of a function display the default values defined for the data type.</p> <p>As long as the function block or function has not been completely processed, the outputs retain the default values defined for the data type.</p> <p>ENO Output: The output is also set to FALSE.</p> </td></tr> </table>	TRUE	<p>The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle.</p> <p>ENO Output: The output is TRUE and can be set to FALSE separately based on the internal error detection.</p>	FALSE	<p>The function block or function is not processed by the user program. The following variables and value field assignment are not assigned.</p> <p>The outputs of a function block display the results of the user program cycle that was lastly processed.</p> <p>The outputs of a function display the default values defined for the data type.</p> <p>As long as the function block or function has not been completely processed, the outputs retain the default values defined for the data type.</p> <p>ENO Output: The output is also set to FALSE.</p>
TRUE	<p>The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle.</p> <p>ENO Output: The output is TRUE and can be set to FALSE separately based on the internal error detection.</p>					
FALSE	<p>The function block or function is not processed by the user program. The following variables and value field assignment are not assigned.</p> <p>The outputs of a function block display the results of the user program cycle that was lastly processed.</p> <p>The outputs of a function display the default values defined for the data type.</p> <p>As long as the function block or function has not been completely processed, the outputs retain the default values defined for the data type.</p> <p>ENO Output: The output is also set to FALSE.</p>					

8.5.2 Outputs

Output	Data type	Description	
OUT	ANY_REAL	Exponential function: The <i>Euler's number</i> is raised to the power of IN. <ul style="list-style-type: none"> • If IN is NaN, OUT is NaN as well. • If IN is +INF, OUT is +INF as well. • If IN is -INF, OUT = 0.0. 	
ENO	BOOL	TRUE	<ul style="list-style-type: none"> • EN = TRUE and • The mathematical function was performed without errors.
		FALSE	<ul style="list-style-type: none"> • EN = FALSE or • Error: The value of IN results in a value exceeding the maximum value of the data type of OUT. The mathematical function could not be performed. The value of OUT is +INF (+infinite).

8.6 SIN (Trigonometric Function: Sine)

The SIN function calculates the sine of the function value in radians.

8.6.1 Inputs

Input	Data type	Description				
IN	ANY_REAL	<p>Input IN</p> <p>Range of values: -2* pi to +2 * pi</p> <p>Exceeding the upper or lower limit of the range of values does not result in an error. However, the accuracy of the results is reduced.</p> <p>This is not an error in SILworX, but results from the finite number of elements in the series expansion of the sine function (→ Sine and Cosine Expansion).</p>				
<u>EN</u>	BOOL	<p>EN is hidden by default. If EN is connected, it is always TRUE.</p> <p>Show EN/ENO: Right-click a function block or function, and select Show EN/ENO from the context menu.</p> <p>As soon as EN or ENO is connected, it can no longer be hidden.</p> <table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td style="width: 25%; padding: 5px;">TRUE</td><td style="width: 75%; padding: 5px;"> The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle. ENO Output: The output is TRUE and can be set to FALSE separately based on the internal error detection. </td></tr> <tr> <td style="padding: 5px;">FALSE</td><td style="padding: 5px;"> The function block or function is not processed by the user program. The following variables and value field assignment are not assigned. The outputs of a function block display the results of the user program cycle that was lastly processed. The outputs of a function display the default values defined for the data type. As long as the function block or function has not been completely processed, the outputs retain the default values defined for the data type. ENO Output: The output is also set to FALSE. </td></tr> </table>	TRUE	The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle. ENO Output: The output is TRUE and can be set to FALSE separately based on the internal error detection.	FALSE	The function block or function is not processed by the user program. The following variables and value field assignment are not assigned. The outputs of a function block display the results of the user program cycle that was lastly processed. The outputs of a function display the default values defined for the data type. As long as the function block or function has not been completely processed, the outputs retain the default values defined for the data type. ENO Output: The output is also set to FALSE.
TRUE	The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle. ENO Output: The output is TRUE and can be set to FALSE separately based on the internal error detection.					
FALSE	The function block or function is not processed by the user program. The following variables and value field assignment are not assigned. The outputs of a function block display the results of the user program cycle that was lastly processed. The outputs of a function display the default values defined for the data type. As long as the function block or function has not been completely processed, the outputs retain the default values defined for the data type. ENO Output: The output is also set to FALSE.					

8.6.2 Outputs

Output	Data type	Description	
OUT	ANY_REAL	<p>Sine of IN in radians</p> <ul style="list-style-type: none">• If IN is NaN, OUT is NaN as well.• If IN is +INF, OUT is NaN.• If IN is -INF, OUT is NaN.	
<u>ENO</u>	BOOL	TRUE	If EN is TRUE and the current cycle of the function block or function was processed without errors, ENO is also TRUE. The function cannot result in an error.
		FALSE	Only if EN is also FALSE or while the function block or function is being processed.

8.7 COS (Trigonometric Function: Cosine)

The COS function calculates the cosine of the function value in radians.

8.7.1 Inputs

Input	Data type	Description				
IN	ANY_REAL	<p>Input IN</p> <p>Range of values: -2* pi to +2 * pi</p> <p>Exceeding the upper or lower limit of the range of values does not result in an error. However, the accuracy of the results is reduced.</p> <p>This is not an error in SILworX, but results from the finite number of elements in the series expansion of the cosine function (→ Sine and Cosine Expansion).</p>				
<u>EN</u>	BOOL	<p>EN is hidden by default. If EN is connected, it is always TRUE.</p> <p>Show EN/ENO: Right-click a function block or function, and select Show EN/ENO from the context menu.</p> <p>As soon as EN or ENO is connected, it can no longer be hidden.</p> <table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td style="width: 25%; padding: 5px;">TRUE</td><td style="width: 75%; padding: 5px;"> The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle. ENO Output: The output is TRUE and can be set to FALSE separately based on the internal error detection. </td></tr> <tr> <td style="padding: 5px;">FALSE</td><td style="padding: 5px;"> The function block or function is not processed by the user program. The following variables and value field assignment are not assigned. The outputs of a function block display the results of the user program cycle that was lastly processed. The outputs of a function display the default values defined for the data type. As long as the function block or function has not been completely processed, the outputs retain the default values defined for the data type. ENO Output: The output is also set to FALSE. </td></tr> </table>	TRUE	The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle. ENO Output: The output is TRUE and can be set to FALSE separately based on the internal error detection.	FALSE	The function block or function is not processed by the user program. The following variables and value field assignment are not assigned. The outputs of a function block display the results of the user program cycle that was lastly processed. The outputs of a function display the default values defined for the data type. As long as the function block or function has not been completely processed, the outputs retain the default values defined for the data type. ENO Output: The output is also set to FALSE.
TRUE	The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle. ENO Output: The output is TRUE and can be set to FALSE separately based on the internal error detection.					
FALSE	The function block or function is not processed by the user program. The following variables and value field assignment are not assigned. The outputs of a function block display the results of the user program cycle that was lastly processed. The outputs of a function display the default values defined for the data type. As long as the function block or function has not been completely processed, the outputs retain the default values defined for the data type. ENO Output: The output is also set to FALSE.					

8.7.2 Outputs

Output	Data type	Description	
OUT	ANY_REAL	<p>Cosine of IN in radians</p> <ul style="list-style-type: none">• If IN is NaN, OUT is NaN as well.• If IN is +INF, OUT is NaN.• If IN is -INF, OUT is NaN.	
<u>ENO</u>	BOOL	TRUE	If EN is TRUE and the current cycle of the function block or function was processed without errors, ENO is also TRUE. The function cannot result in an error.
		FALSE	Only if EN is also FALSE or while the function block or function is being processed.

8.8 TAN (Trigonometric Function: Tangent)

The TAN function calculates the tangent of the function value in radians.

8.8.1 Inputs

Input	Data type	Description				
IN	ANY_REAL	<p>Input IN</p> <p>Range of values: -2* pi to +2 * pi</p> <p>Exceeding the upper or lower limit of the range of values does not result in an error. However, the accuracy of the results is reduced.</p> <p>This is not an error in SILworX, but results from the finite number of elements in the series expansion of the sine function (→ Sine and Cosine Expansion).</p>				
EN	BOOL	<p>EN is hidden by default. If EN is connected, it is always TRUE.</p> <p>Show EN/ENO: Right-click a function block or function, and select Show EN/ENO from the context menu.</p> <p>As soon as EN or ENO is connected, it can no longer be hidden.</p> <table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td style="width: 25%; padding: 5px;">TRUE</td><td style="width: 75%; padding: 5px;"> The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle. ENO Output: The output is TRUE and can be set to FALSE separately based on the internal error detection. </td></tr> <tr> <td style="padding: 5px;">FALSE</td><td style="padding: 5px;"> The function block or function is not processed by the user program. The following variables and value field assignment are not assigned. The outputs of a function block display the results of the user program cycle that was lastly processed. The outputs of a function display the default values defined for the data type. As long as the function block or function has not been completely processed, the outputs retain the default values defined for the data type. ENO Output: The output is also set to FALSE. </td></tr> </table>	TRUE	The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle. ENO Output: The output is TRUE and can be set to FALSE separately based on the internal error detection.	FALSE	The function block or function is not processed by the user program. The following variables and value field assignment are not assigned. The outputs of a function block display the results of the user program cycle that was lastly processed. The outputs of a function display the default values defined for the data type. As long as the function block or function has not been completely processed, the outputs retain the default values defined for the data type. ENO Output: The output is also set to FALSE.
TRUE	The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle. ENO Output: The output is TRUE and can be set to FALSE separately based on the internal error detection.					
FALSE	The function block or function is not processed by the user program. The following variables and value field assignment are not assigned. The outputs of a function block display the results of the user program cycle that was lastly processed. The outputs of a function display the default values defined for the data type. As long as the function block or function has not been completely processed, the outputs retain the default values defined for the data type. ENO Output: The output is also set to FALSE.					

8.8.2 Outputs

Output	Data type	Description	
OUT	ANY_REAL	<p>OUT = tan(IN) in radians.</p> <ul style="list-style-type: none">• If IN is NaN, OUT is NaN as well.• If IN is +INF, OUT is NaN.• If IN is -INF, OUT is NaN.	
ENO	BOOL	TRUE	If EN is TRUE and the current cycle of the function block or function was processed without errors, ENO is also TRUE. The function cannot result in an error.
		FALSE	Only if EN is also FALSE or while the function block or function is being processed.

8.9 ASIN (Trigonometric Function: Arcus Sine)

The ASIN function calculates the arc sine of the function value in radians.

8.9.1 Inputs

Input	Data type	Description				
IN	ANY_REAL	<p>Input IN</p> <p>Range of values: $-1.0 \leq IN \leq 1.0$</p>				
<u>EN</u>	BOOL	<p>EN is hidden by default. If EN is connected, it is always TRUE.</p> <p>Show EN/ENO: Right-click a function block or function, and select Show EN/ENO from the context menu.</p> <p>As soon as EN or ENO is connected, it can no longer be hidden.</p> <table border="1"> <tr> <td>TRUE</td><td> <p>The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle.</p> <p>ENO Output: The output is TRUE and can be set to FALSE separately based on the internal error detection.</p> </td></tr> <tr> <td>FALSE</td><td> <p>The function block or function is not processed by the user program. The following variables and value field assignment are not assigned.</p> <p>The outputs of a function block display the results of the user program cycle that was lastly processed.</p> <p>The outputs of a function display the default values defined for the data type.</p> <p>As long as the function block or function has not been completely processed, the outputs retain the default values defined for the data type.</p> <p>ENO Output: The output is also set to FALSE.</p> </td></tr> </table>	TRUE	<p>The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle.</p> <p>ENO Output: The output is TRUE and can be set to FALSE separately based on the internal error detection.</p>	FALSE	<p>The function block or function is not processed by the user program. The following variables and value field assignment are not assigned.</p> <p>The outputs of a function block display the results of the user program cycle that was lastly processed.</p> <p>The outputs of a function display the default values defined for the data type.</p> <p>As long as the function block or function has not been completely processed, the outputs retain the default values defined for the data type.</p> <p>ENO Output: The output is also set to FALSE.</p>
TRUE	<p>The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle.</p> <p>ENO Output: The output is TRUE and can be set to FALSE separately based on the internal error detection.</p>					
FALSE	<p>The function block or function is not processed by the user program. The following variables and value field assignment are not assigned.</p> <p>The outputs of a function block display the results of the user program cycle that was lastly processed.</p> <p>The outputs of a function display the default values defined for the data type.</p> <p>As long as the function block or function has not been completely processed, the outputs retain the default values defined for the data type.</p> <p>ENO Output: The output is also set to FALSE.</p>					

8.9.2 Outputs

Output	Data type	Description	
OUT	ANY_REAL	<p>Arc sine of IN in radians</p> <ul style="list-style-type: none">• If IN is NaN, OUT is NaN as well.• If IN is +INF, OUT is NaN.• If IN is -INF, OUT is NaN.	
<u>ENO</u>	BOOL	TRUE	If EN = TRUE and the function was processed without errors.
		FALSE	<ul style="list-style-type: none">• If also EN = FALSE or• Error: IN is out of range. The value of OUT is NaN (not a number).

8.10 ACOS (Trigonometric Function: Arc Cosine)

The ACOS function calculates the arc cosine of the function value in radians.

8.10.1 Inputs

Input	Data type	Description				
IN	ANY_REAL	<p>Input IN</p> <p>Range of values: $-1.0 \leq IN \leq 1.0$</p>				
<u>EN</u>	BOOL	<p>EN is hidden by default. If EN is connected, it is always TRUE.</p> <p>Show EN/ENO: Right-click a function block or function, and select Show EN/ENO from the context menu.</p> <p>As soon as EN or ENO is connected, it can no longer be hidden.</p> <table border="1"> <tr> <td>TRUE</td><td> <p>The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle.</p> <p>ENO Output: The output is TRUE and can be set to FALSE separately based on the internal error detection.</p> </td></tr> <tr> <td>FALSE</td><td> <p>The function block or function is not processed by the user program. The following variables and value field assignment are not assigned.</p> <p>The outputs of a function block display the results of the user program cycle that was lastly processed.</p> <p>The outputs of a function display the default values defined for the data type.</p> <p>As long as the function block or function has not been completely processed, the outputs retain the default values defined for the data type.</p> <p>ENO Output: The output is also set to FALSE.</p> </td></tr> </table>	TRUE	<p>The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle.</p> <p>ENO Output: The output is TRUE and can be set to FALSE separately based on the internal error detection.</p>	FALSE	<p>The function block or function is not processed by the user program. The following variables and value field assignment are not assigned.</p> <p>The outputs of a function block display the results of the user program cycle that was lastly processed.</p> <p>The outputs of a function display the default values defined for the data type.</p> <p>As long as the function block or function has not been completely processed, the outputs retain the default values defined for the data type.</p> <p>ENO Output: The output is also set to FALSE.</p>
TRUE	<p>The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle.</p> <p>ENO Output: The output is TRUE and can be set to FALSE separately based on the internal error detection.</p>					
FALSE	<p>The function block or function is not processed by the user program. The following variables and value field assignment are not assigned.</p> <p>The outputs of a function block display the results of the user program cycle that was lastly processed.</p> <p>The outputs of a function display the default values defined for the data type.</p> <p>As long as the function block or function has not been completely processed, the outputs retain the default values defined for the data type.</p> <p>ENO Output: The output is also set to FALSE.</p>					

8.10.2 Outputs

Output	Data type	Description
OUT	ANY_REAL	Arc cosine of IN in radians <ul style="list-style-type: none">• If IN is NaN, OUT is NaN as well.• If IN is +INF, OUT is NaN.• If IN is -INF, OUT is NaN.
<u>ENO</u>	BOOL	TRUE If EN = TRUE and the function was processed without errors.
		FALSE <ul style="list-style-type: none">• If also EN = FALSE or• Error: IN is out of range. The value of OUT is NaN (not a number).

8.11 ATAN (Trigonometric Function: Arc Tangent)

The ATAN function calculates the arc tangent of the function value in radians.

8.11.1 Inputs

Input	Data type	Description				
IN	ANY_REAL	<p>Input IN</p> <p>Range of values: $-\infty < \text{IN} < \infty$</p>				
<u>EN</u>	BOOL	<p>EN is hidden by default. If EN is connected, it is always TRUE.</p> <p>Show EN/ENO: Right-click a function block or function, and select Show EN/ENO from the context menu.</p> <p>As soon as EN or ENO is connected, it can no longer be hidden.</p> <table border="1"> <tr> <td>TRUE</td><td> <p>The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle.</p> <p>ENO Output: The output is TRUE and can be set to FALSE separately based on the internal error detection.</p> </td></tr> <tr> <td>FALSE</td><td> <p>The function block or function is not processed by the user program. The following variables and value field assignment are not assigned.</p> <p>The outputs of a function block display the results of the user program cycle that was lastly processed.</p> <p>The outputs of a function display the default values defined for the data type.</p> <p>As long as the function block or function has not been completely processed, the outputs retain the default values defined for the data type.</p> <p>ENO Output: The output is also set to FALSE.</p> </td></tr> </table>	TRUE	<p>The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle.</p> <p>ENO Output: The output is TRUE and can be set to FALSE separately based on the internal error detection.</p>	FALSE	<p>The function block or function is not processed by the user program. The following variables and value field assignment are not assigned.</p> <p>The outputs of a function block display the results of the user program cycle that was lastly processed.</p> <p>The outputs of a function display the default values defined for the data type.</p> <p>As long as the function block or function has not been completely processed, the outputs retain the default values defined for the data type.</p> <p>ENO Output: The output is also set to FALSE.</p>
TRUE	<p>The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle.</p> <p>ENO Output: The output is TRUE and can be set to FALSE separately based on the internal error detection.</p>					
FALSE	<p>The function block or function is not processed by the user program. The following variables and value field assignment are not assigned.</p> <p>The outputs of a function block display the results of the user program cycle that was lastly processed.</p> <p>The outputs of a function display the default values defined for the data type.</p> <p>As long as the function block or function has not been completely processed, the outputs retain the default values defined for the data type.</p> <p>ENO Output: The output is also set to FALSE.</p>					

8.11.2 Outputs

Output	Data type	Description	
OUT	ANY_REAL	<p>Arc tangent of IN in radians.</p> <ul style="list-style-type: none">• If IN is NaN, OUT is NaN as well.• If IN is +INF, OUT is pi/2.• If IN is -INF, OUT is -pi/2.	
<u>ENO</u>	BOOL	TRUE	If EN is TRUE and the current cycle of the function block or function was processed without errors, ENO is also TRUE. The function cannot result in an error.
		FALSE	Only if EN is also FALSE or while the function block or function is being processed.

8.12 ADD (Addition)

The ADD function adds the values of up to 16 inputs.

8.12.1 Inputs

Input	Data type	Description				
IN1	ANY_NUM (*)	Input IN1				
IN2	ANY_NUM (*)	Input IN2				
Extendible	ANY_NUM (*)	Up to IN16 (*) All inputs must be of the same data type.				
<u>EN</u>	BOOL	<p>EN is hidden by default. If EN is connected, it is always TRUE.</p> <p>Show EN/ENO: Right-click a function block or function, and select Show EN/ENO from the context menu.</p> <p>As soon as EN or ENO is connected, it can no longer be hidden.</p> <table border="1"> <tr> <td>TRUE</td><td> <p>The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle.</p> <p>ENO Output: The output is TRUE and can be set to FALSE separately based on the internal error detection.</p> </td></tr> <tr> <td>FALSE</td><td> <p>The function block or function is not processed by the user program. The following variables and value field assignment are not assigned.</p> <p>The outputs of a function block display the results of the user program cycle that was lastly processed.</p> <p>The outputs of a function display the default values defined for the data type.</p> <p>As long as the function block or function has not been completely processed, the outputs retain the default values defined for the data type.</p> <p>ENO Output: The output is also set to FALSE.</p> </td></tr> </table>	TRUE	<p>The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle.</p> <p>ENO Output: The output is TRUE and can be set to FALSE separately based on the internal error detection.</p>	FALSE	<p>The function block or function is not processed by the user program. The following variables and value field assignment are not assigned.</p> <p>The outputs of a function block display the results of the user program cycle that was lastly processed.</p> <p>The outputs of a function display the default values defined for the data type.</p> <p>As long as the function block or function has not been completely processed, the outputs retain the default values defined for the data type.</p> <p>ENO Output: The output is also set to FALSE.</p>
TRUE	<p>The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle.</p> <p>ENO Output: The output is TRUE and can be set to FALSE separately based on the internal error detection.</p>					
FALSE	<p>The function block or function is not processed by the user program. The following variables and value field assignment are not assigned.</p> <p>The outputs of a function block display the results of the user program cycle that was lastly processed.</p> <p>The outputs of a function display the default values defined for the data type.</p> <p>As long as the function block or function has not been completely processed, the outputs retain the default values defined for the data type.</p> <p>ENO Output: The output is also set to FALSE.</p>					

8.12.2 Outputs

Output	Data type	Description																																	
OUT	ANY_NUM	<p>Sum of the values from IN1 to INn.</p> <p>If the range limits for OUT are exceeded, the overflow is signaled on ENO and output to OUT 1.</p> <p>Examples:</p> <table border="1"> <thead> <tr> <th>Data type</th> <th>Function</th> <th>OUT</th> <th>Overflow</th> </tr> </thead> <tbody> <tr> <td rowspan="4"><u>SINT</u></td> <td>ADD(60; 60)</td> <td>120</td> <td></td> </tr> <tr> <td>ADD(66; 66)</td> <td>-124</td> <td>X</td> </tr> <tr> <td>ADD(127; 127; 127; 127)</td> <td>-4</td> <td>X</td> </tr> <tr> <td>ADD(65; 65; 65; 65)</td> <td>4</td> <td>X</td> </tr> <tr> <td rowspan="4"><u>REAL</u></td> <td>ADD(1.7e+38; 1.7e+38)</td> <td>3.4e+38</td> <td></td> </tr> <tr> <td>ADD(1.8e+38; 1.8e+38)</td> <td>INF</td> <td>X</td> </tr> <tr> <td>ADD(-3.4e+38; -3.4e+38)</td> <td>-INF</td> <td>X</td> </tr> <tr> <td>ADD(NaN; 1.0)</td> <td>NaN</td> <td>X</td> </tr> </tbody> </table>				Data type	Function	OUT	Overflow	<u>SINT</u>	ADD(60; 60)	120		ADD(66; 66)	-124	X	ADD(127; 127; 127; 127)	-4	X	ADD(65; 65; 65; 65)	4	X	<u>REAL</u>	ADD(1.7e+38; 1.7e+38)	3.4e+38		ADD(1.8e+38; 1.8e+38)	INF	X	ADD(-3.4e+38; -3.4e+38)	-INF	X	ADD(NaN; 1.0)	NaN	X
Data type	Function	OUT	Overflow																																
<u>SINT</u>	ADD(60; 60)	120																																	
	ADD(66; 66)	-124	X																																
	ADD(127; 127; 127; 127)	-4	X																																
	ADD(65; 65; 65; 65)	4	X																																
<u>REAL</u>	ADD(1.7e+38; 1.7e+38)	3.4e+38																																	
	ADD(1.8e+38; 1.8e+38)	INF	X																																
	ADD(-3.4e+38; -3.4e+38)	-INF	X																																
	ADD(NaN; 1.0)	NaN	X																																
<u>ENO</u>	BOOL	TRUE	If EN = TRUE and the function was processed without errors.																																
		FALSE	<ul style="list-style-type: none"> • If also EN = FALSE or • Error: The range of values of OUT is exceeded If the data type of IN is an integer (ANY_INT), the overflow value is output to OUT. If the data type of IN is REAL or LREAL, the the value of OUT is either NaN or +INF/-INF. 																																

8.12.3 Implementation in Structured Text (ST)

```
OUT := IN1 + IN2 + ... + INn;
```

8.13 MUL (Multiplication)

The MUL function multiplies the values of up to 16 inputs.

8.13.1 Inputs

Input	Data type	Description				
IN1	ANY_NUM (*)	Input IN1				
IN2	ANY_NUM (*)	Input IN2				
Extendible	ANY_NUM (*)	Up to IN16 (*) All inputs must be of the same data type.				
<u>EN</u>	BOOL	<p>EN is hidden by default. If EN is connected, it is always TRUE.</p> <p>Show EN/ENO: Right-click a function block or function, and select Show EN/ENO from the context menu.</p> <p>As soon as EN or ENO is connected, it can no longer be hidden.</p> <table border="1"> <tr> <td>TRUE</td><td> <p>The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle.</p> <p>ENO Output: The output is TRUE and can be set to FALSE separately based on the internal error detection.</p> </td></tr> <tr> <td>FALSE</td><td> <p>The function block or function is not processed by the user program. The following variables and value field assignment are not assigned.</p> <p>The outputs of a function block display the results of the user program cycle that was lastly processed.</p> <p>The outputs of a function display the default values defined for the data type.</p> <p>As long as the function block or function has not been completely processed, the outputs retain the default values defined for the data type.</p> <p>ENO Output: The output is also set to FALSE.</p> </td></tr> </table>	TRUE	<p>The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle.</p> <p>ENO Output: The output is TRUE and can be set to FALSE separately based on the internal error detection.</p>	FALSE	<p>The function block or function is not processed by the user program. The following variables and value field assignment are not assigned.</p> <p>The outputs of a function block display the results of the user program cycle that was lastly processed.</p> <p>The outputs of a function display the default values defined for the data type.</p> <p>As long as the function block or function has not been completely processed, the outputs retain the default values defined for the data type.</p> <p>ENO Output: The output is also set to FALSE.</p>
TRUE	<p>The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle.</p> <p>ENO Output: The output is TRUE and can be set to FALSE separately based on the internal error detection.</p>					
FALSE	<p>The function block or function is not processed by the user program. The following variables and value field assignment are not assigned.</p> <p>The outputs of a function block display the results of the user program cycle that was lastly processed.</p> <p>The outputs of a function display the default values defined for the data type.</p> <p>As long as the function block or function has not been completely processed, the outputs retain the default values defined for the data type.</p> <p>ENO Output: The output is also set to FALSE.</p>					

8.13.2 Outputs

Output	Data type	Description			
		<p>$OUT = IN1 * IN2 * \dots * INn$</p> <p>If the range limits for OUT are exceeded, the overflow is signaled on ENO and output to OUT 1. For ANY_INT data types, the overflow is calculated based on the two's complement.</p>			
Examples:					
		Data type	Function	OUT	Overflow
OUT	ANY_NUM	<u>SINT</u>	MUL(20; 5)	100	
			MUL(20; 7)	-116	X
			MUL(-128; -1)	-128	X
			MUL(127; -127)	-1	X
		<u>REAL</u>	MUL(1.5e+09; -2.0e+03)	3.0e+12	
			MUL(3.4e+38; 2.0)	INF	X
			MUL(1.0e+38; -4.0)	-INF	X
			MUL(NaN; 1.0)	NaN	X
<u>ENO</u>	BOOL	TRUE	<ul style="list-style-type: none"> EN = TRUE and The mathematical function was performed without errors. 		
		FALSE	<ul style="list-style-type: none"> EN = FALSE or Error: Result overflow. 		

8.13.3 Implementation in Structured Text (ST)

```
OUT := IN1 * IN2 * .. * INn;
```

8.14 SUB (Subtraction)

The SUB function subtracts the value of IN2 from the value of IN1.

8.14.1 Inputs

Input	Data type	Description				
IN1	ANY_NUM (*)	Input IN1				
IN2	ANY_NUM (*)	Input IN2 (*) All inputs must be of the same data type.				
<u>EN</u>	BOOL	<p>EN is hidden by default. If EN is connected, it is always TRUE. Show EN/ENO: Right-click a function block or function, and select Show EN/ENO from the context menu.</p> <p>As soon as EN or ENO is connected, it can no longer be hidden.</p> <table border="1"> <tr> <td>TRUE</td><td> <p>The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle.</p> <p>ENO Output: The output is TRUE and can be set to FALSE separately based on the internal error detection.</p> </td></tr> <tr> <td>FALSE</td><td> <p>The function block or function is not processed by the user program. The following variables and value field assignment are not assigned.</p> <p>The outputs of a function block display the results of the user program cycle that was lastly processed.</p> <p>The outputs of a function display the default values defined for the data type.</p> <p>As long as the function block or function has not been completely processed, the outputs retain the default values defined for the data type.</p> <p>ENO Output: The output is also set to FALSE.</p> </td></tr> </table>	TRUE	<p>The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle.</p> <p>ENO Output: The output is TRUE and can be set to FALSE separately based on the internal error detection.</p>	FALSE	<p>The function block or function is not processed by the user program. The following variables and value field assignment are not assigned.</p> <p>The outputs of a function block display the results of the user program cycle that was lastly processed.</p> <p>The outputs of a function display the default values defined for the data type.</p> <p>As long as the function block or function has not been completely processed, the outputs retain the default values defined for the data type.</p> <p>ENO Output: The output is also set to FALSE.</p>
TRUE	<p>The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle.</p> <p>ENO Output: The output is TRUE and can be set to FALSE separately based on the internal error detection.</p>					
FALSE	<p>The function block or function is not processed by the user program. The following variables and value field assignment are not assigned.</p> <p>The outputs of a function block display the results of the user program cycle that was lastly processed.</p> <p>The outputs of a function display the default values defined for the data type.</p> <p>As long as the function block or function has not been completely processed, the outputs retain the default values defined for the data type.</p> <p>ENO Output: The output is also set to FALSE.</p>					

8.14.2 Outputs

Output	Data type	Description			
OUT	ANY_NUM	<p>OUT = IN1 - IN2</p> <p>If the lower range limit for OUT is exceeded, the overflow is signaled on ENO and output to OUT.</p>			
Examples:					
Data type	Function	OUT	Overflow		
SINT	SUB(60; 50)	10			
	SUB(60; 70)	-10			
	SUB(-128; -3)	-125	X		
	SUB(127; -127)	-2	X		
REAL	SUB(1.5e+09; 0.5e+09)	1.0e+09			
	SUB(3.4e+38; -0.5e+38)	INF	X		
	SUB(-1.8e+38; 1.8e+38)	-INF	X		
	SUB(NaN; 1.0)	NaN	X		
ENO	TRUE	If EN = TRUE and the function was processed without errors.			
	FALSE	<ul style="list-style-type: none"> If also EN = FALSE or Error: The range of values of OUT is exceeded If the data type of IN is an integer (ANY_INT), the overflow value is output to OUT. If the data type of IN is REAL or LREAL, the the value of OUT is either NaN or +INF/-INF. 			

8.14.3 Implementation in Structured Text (ST)

```
OUT := IN1 - IN2;
```

8.15 DIV (Division)

8.15.1 Inputs

Input	Data type	Description		
IN1	ANY_NUM	Input IN1 (dividend)		
IN2	Same as with IN1	Input IN2 (divisor)		
EN	BOOL	<p>EN is hidden by default. If EN is connected, it is always TRUE.</p> <p>Show EN/ENO: Right-click a function block or function, and select Show EN/ENO from the context menu.</p> <p>As soon as EN or ENO is connected, it can no longer be hidden.</p> <table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td style="padding: 5px; vertical-align: top;"> TRUE The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle. ENO Output: The output is TRUE and can be set to FALSE separately based on the internal error detection. </td> <td style="padding: 5px; vertical-align: top;"> FALSE The function block or function is not processed by the user program. The following variables and value field assignment are not assigned. The outputs of a function block display the results of the user program cycle that was lastly processed. The outputs of a function display the default values defined for the data type. As long as the function block or function has not been completely processed, the outputs retain the default values defined for the data type. ENO Output: The output is also set to FALSE. </td> </tr> </table>	TRUE The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle. ENO Output: The output is TRUE and can be set to FALSE separately based on the internal error detection.	FALSE The function block or function is not processed by the user program. The following variables and value field assignment are not assigned. The outputs of a function block display the results of the user program cycle that was lastly processed. The outputs of a function display the default values defined for the data type. As long as the function block or function has not been completely processed, the outputs retain the default values defined for the data type. ENO Output: The output is also set to FALSE.
TRUE The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle. ENO Output: The output is TRUE and can be set to FALSE separately based on the internal error detection.	FALSE The function block or function is not processed by the user program. The following variables and value field assignment are not assigned. The outputs of a function block display the results of the user program cycle that was lastly processed. The outputs of a function display the default values defined for the data type. As long as the function block or function has not been completely processed, the outputs retain the default values defined for the data type. ENO Output: The output is also set to FALSE.			

8.15.2 Outputs

Output	Data type	Description																					
OUT	Same as with IN1	$OUT = IN1 / IN2$ Behavior with division by 0 (IN2 = 0) <table border="1" style="margin-left: auto; margin-right: auto; border-collapse: collapse;"> <tr> <th>Data type</th><th>IN1</th><th>OUT</th><th>Example</th></tr> <tr> <td>ANY_INT</td><td>IN1 = Integer</td><td>OUT = 0</td><td>$4 / 0 = 0$</td></tr> <tr> <td rowspan="3">ANY_REAL</td><td>IN1 = negative</td><td>OUT = -INF</td><td>$-1.0 / 0 = -INF$</td></tr> <tr> <td>IN1 = 0.0</td><td>OUT = NaN</td><td>$0.0 / 0 = NaN$</td></tr> <tr> <td>IN1 = positive</td><td>OUT = +INF</td><td>$1.0 / 0 = +INF$</td></tr> </table>				Data type	IN1	OUT	Example	ANY_INT	IN1 = Integer	OUT = 0	$4 / 0 = 0$	ANY_REAL	IN1 = negative	OUT = -INF	$-1.0 / 0 = -INF$	IN1 = 0.0	OUT = NaN	$0.0 / 0 = NaN$	IN1 = positive	OUT = +INF	$1.0 / 0 = +INF$
Data type	IN1	OUT	Example																				
ANY_INT	IN1 = Integer	OUT = 0	$4 / 0 = 0$																				
ANY_REAL	IN1 = negative	OUT = -INF	$-1.0 / 0 = -INF$																				
	IN1 = 0.0	OUT = NaN	$0.0 / 0 = NaN$																				
	IN1 = positive	OUT = +INF	$1.0 / 0 = +INF$																				

Output	Data type	Description				
OUT	Same as with IN1	Behavior upon overflow				
		Data type	IN1	OUT	Example	
		ANY_INT	IN1 = <u>SINT</u>	OUT = Overflow	-128 / -1 = -128	
		ANY_REAL	IN1 = REAL	OUT = +INF/-INF		
<u>ENO</u>	BOOL	TRUE	<ul style="list-style-type: none"> EN = TRUE and The mathematical function was performed without errors. 			
		FALSE	<ul style="list-style-type: none"> EN = FALSE or Error: The mathematical function was performed with error (overflow or division by 0). 			

8.15.3 Behavior with REAL and LREAL

		IN1					
		>0 (<∞)	<0 (>-∞)	∞	-∞	NaN	0.0
IN2	>0 (<∞)	nn	nn	∞	-∞	NaN	0.0
	<0 (>-∞)	nn	nn	-∞	∞	NaN	0.0
	∞	0.0	0.0	NaN	NaN	NaN	0.0
	-∞	0.0	0.0	NaN	NaN	NaN	0.0
	nan	NaN	NaN	NaN	NaN	NaN	NaN
	0.0	< ∞	>-∞	∞	-∞	NaN	NaN

	ENO = TRUE
	ENO = FALSE
nn	Calculated value IN1/IN2
∞	Infinite
NaN	Not a Number (invalid REAL value)

8.15.4 Implementation in Structured Text (ST)

```
OUT := IN1 / IN2;
```

8.16 MOD (Modulo)

8.16.1 Inputs

Input	Data type	Description				
IN1	ANY_INT	Input IN1				
IN2	ANY_INT	Input IN2				
<u>EN</u>	BOOL	<p>EN is hidden by default. If EN is connected, it is always TRUE.</p> <p>Show EN/ENO: Right-click a function block or function, and select Show EN/ENO from the context menu.</p> <p>As soon as EN or ENO is connected, it can no longer be hidden.</p> <table border="1"> <tr> <td>TRUE</td><td> <p>The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle.</p> <p>ENO Output: The output is TRUE and can be set to FALSE separately based on the internal error detection.</p> </td></tr> <tr> <td>FALSE</td><td> <p>The function block or function is not processed by the user program. The following variables and value field assignment are not assigned.</p> <p>The outputs of a function block display the results of the user program cycle that was lastly processed.</p> <p>The outputs of a function display the default values defined for the data type.</p> <p>As long as the function block or function has not been completely processed, the outputs retain the default values defined for the data type.</p> <p>ENO Output: The output is also set to FALSE.</p> </td></tr> </table>	TRUE	<p>The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle.</p> <p>ENO Output: The output is TRUE and can be set to FALSE separately based on the internal error detection.</p>	FALSE	<p>The function block or function is not processed by the user program. The following variables and value field assignment are not assigned.</p> <p>The outputs of a function block display the results of the user program cycle that was lastly processed.</p> <p>The outputs of a function display the default values defined for the data type.</p> <p>As long as the function block or function has not been completely processed, the outputs retain the default values defined for the data type.</p> <p>ENO Output: The output is also set to FALSE.</p>
TRUE	<p>The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle.</p> <p>ENO Output: The output is TRUE and can be set to FALSE separately based on the internal error detection.</p>					
FALSE	<p>The function block or function is not processed by the user program. The following variables and value field assignment are not assigned.</p> <p>The outputs of a function block display the results of the user program cycle that was lastly processed.</p> <p>The outputs of a function display the default values defined for the data type.</p> <p>As long as the function block or function has not been completely processed, the outputs retain the default values defined for the data type.</p> <p>ENO Output: The output is also set to FALSE.</p>					

8.16.2 Outputs

Output	Data type	Description			
		OUT1 = Integer remainder of IN1/IN2			
Examples:					
		Data type	Function	OUT	Remark
OUT	ANY_INT	ANY_INT	MOD(20; 3)	2	-/-
			MOD(23; -8)	7	-/-
			MOD(32767; 16384)	16383	-/-
			MOD(20; 0)	0	ENO = FALSE
<u>ENO</u>	BOOL	TRUE	If EN = TRUE and the function was processed without errors.		
		FALSE	<ul style="list-style-type: none"> • If also EN = FALSE or • Error: MOD(IN1; 0) → OUT = 0 		

8.16.3 Implementation in Structured Text (ST)

```
OUT := IN1 modulo IN2;
```

8.17 EXPT (Exponentiate)

The EXPT function raises the basis IN1 to the power of the exponent IN2.

8.17.1 Inputs

Input	Data type	Description				
IN1	ANY_REAL	Input IN1				
IN2	ANY_NUM	Input IN2				
<u>EN</u>	BOOL	<p>EN is hidden by default. If EN is connected, it is always TRUE.</p> <p>Show EN/ENO: Right-click a function block or function, and select Show EN/ENO from the context menu.</p> <p>As soon as EN or ENO is connected, it can no longer be hidden.</p> <table border="1"> <tr> <td>TRUE</td><td> <p>The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle.</p> <p>ENO Output: The output is TRUE and can be set to FALSE separately based on the internal error detection.</p> </td></tr> <tr> <td>FALSE</td><td> <p>The function block or function is not processed by the user program. The following variables and value field assignment are not assigned.</p> <p>The outputs of a function block display the results of the user program cycle that was lastly processed.</p> <p>The outputs of a function display the default values defined for the data type.</p> <p>As long as the function block or function has not been completely processed, the outputs retain the default values defined for the data type.</p> <p>ENO Output: The output is also set to FALSE.</p> </td></tr> </table>	TRUE	<p>The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle.</p> <p>ENO Output: The output is TRUE and can be set to FALSE separately based on the internal error detection.</p>	FALSE	<p>The function block or function is not processed by the user program. The following variables and value field assignment are not assigned.</p> <p>The outputs of a function block display the results of the user program cycle that was lastly processed.</p> <p>The outputs of a function display the default values defined for the data type.</p> <p>As long as the function block or function has not been completely processed, the outputs retain the default values defined for the data type.</p> <p>ENO Output: The output is also set to FALSE.</p>
TRUE	<p>The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle.</p> <p>ENO Output: The output is TRUE and can be set to FALSE separately based on the internal error detection.</p>					
FALSE	<p>The function block or function is not processed by the user program. The following variables and value field assignment are not assigned.</p> <p>The outputs of a function block display the results of the user program cycle that was lastly processed.</p> <p>The outputs of a function display the default values defined for the data type.</p> <p>As long as the function block or function has not been completely processed, the outputs retain the default values defined for the data type.</p> <p>ENO Output: The output is also set to FALSE.</p>					

8.17.2 Outputs

Output	Data type	Description
OUT	ANY_REAL	OUT1 = IN1 raised to the power of IN2. The data type for OUT is defined by the data type of IN1.
<u>ENO</u>	BOOL	TRUE <ul style="list-style-type: none">• EN = TRUE and• The mathematical function was performed without errors.
		FALSE <ul style="list-style-type: none">• EN = FALSE or• Error: Result overflow or invalid value (+INF, -INF, NaN).

8.17.3 Implementation in Structured Text (ST)

```
OUT := IN1**IN2;
```

8.18 MOVE

With the MOVE function, the value of IN is controlled and moved to OUT using EN.

8.18.1 Inputs

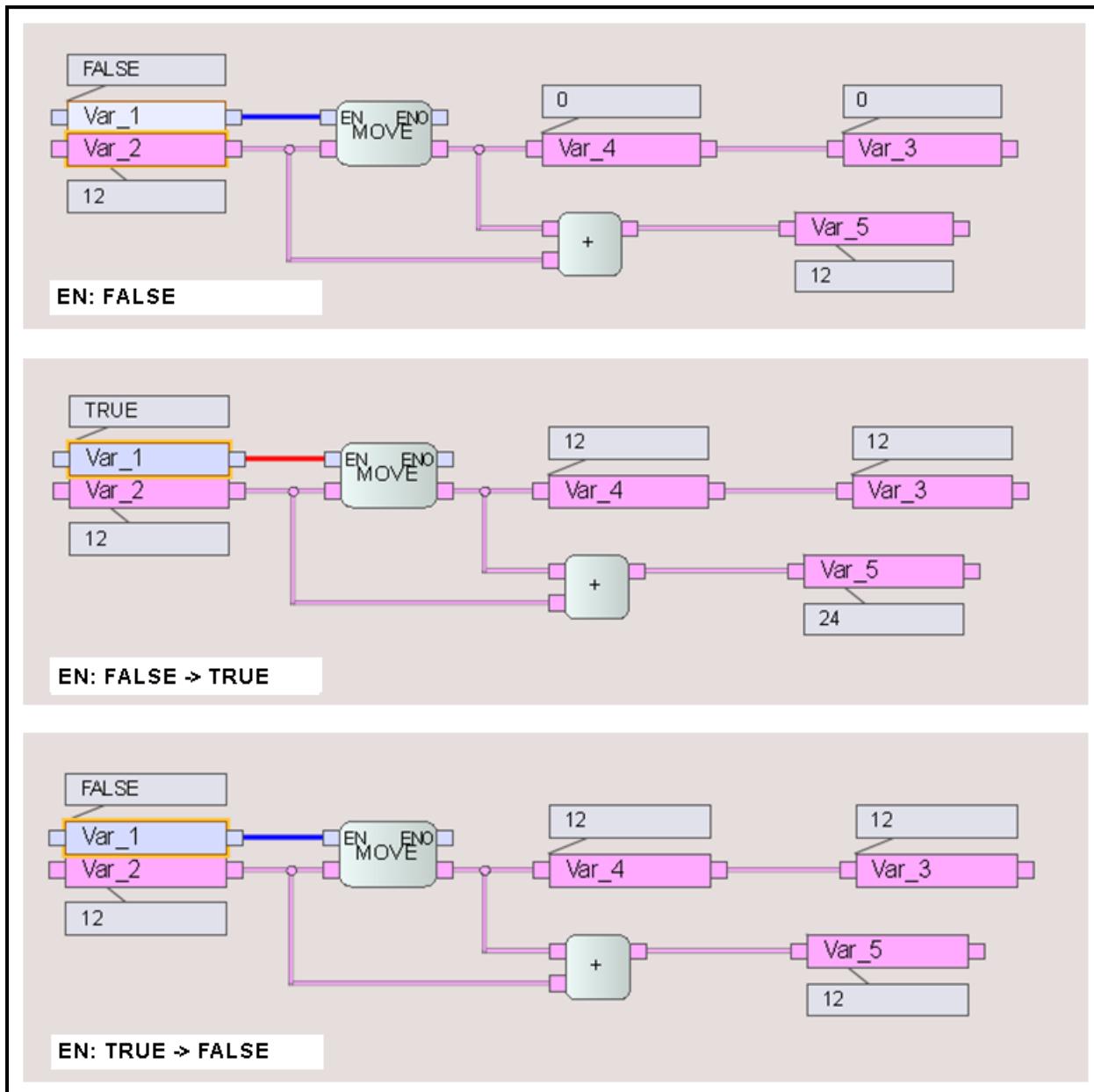
Input	Data type	Description				
IN	ANY	Input IN				
<u>EN</u>	BOOL	<p>EN is hidden by default. If EN is connected, it is always TRUE.</p> <p>Show EN/ENO: Right-click a function block or function, and select Show EN/ENO from the context menu.</p> <p>As soon as EN or ENO is connected, it can no longer be hidden.</p> <table border="1"> <tr> <td>TRUE</td><td>The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle. ENO Output: The output is TRUE and can be set to FALSE separately based on the internal error detection.</td></tr> <tr> <td>FALSE</td><td>The function block or function is not processed by the user program. The following variables and value field assignment are not assigned. The outputs of a function block display the results of the user program cycle that was lastly processed. The outputs of a function display the default values defined for the data type. As long as the function block or function has not been completely processed, the outputs retain the default values defined for the data type. ENO Output: The output is also set to FALSE.</td></tr> </table>	TRUE	The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle. ENO Output: The output is TRUE and can be set to FALSE separately based on the internal error detection.	FALSE	The function block or function is not processed by the user program. The following variables and value field assignment are not assigned. The outputs of a function block display the results of the user program cycle that was lastly processed. The outputs of a function display the default values defined for the data type. As long as the function block or function has not been completely processed, the outputs retain the default values defined for the data type. ENO Output: The output is also set to FALSE.
TRUE	The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle. ENO Output: The output is TRUE and can be set to FALSE separately based on the internal error detection.					
FALSE	The function block or function is not processed by the user program. The following variables and value field assignment are not assigned. The outputs of a function block display the results of the user program cycle that was lastly processed. The outputs of a function display the default values defined for the data type. As long as the function block or function has not been completely processed, the outputs retain the default values defined for the data type. ENO Output: The output is also set to FALSE.					

8.18.2 Outputs

Output	Data type	Description	
OUT	ANY	Use EN to control the values output to OUT.	
<u>ENO</u>	BOOL	EN	OUT
		TRUE	The value of IN is output to OUT.
		FALSE	OUT contains the default value of the data type defined for IN.
		TRUE	If EN is TRUE and the current cycle of the function block or function was processed without errors, ENO is also TRUE. The function cannot result in an error.
		FALSE	Only if EN is also FALSE or while the function block or function is being processed.

8.18.3 Application Example

Notice that MOVE is a function (and not a function block); the output is thus set to the default value of the data type if EN = FALSE.



8.19 CHS (Change Sign)

The CHS function displays the value of IN with changed sign to OUT.

8.19.1 Inputs

Input	Data type	Description		
IN	ANY_NUM	Input IN		
EN	BOOL	<p>EN is hidden by default. If EN is connected, it is always TRUE. Show EN/ENO: Right-click a function block or function, and select Show EN/ENO from the context menu.</p> <p>As soon as EN or ENO is connected, it can no longer be hidden.</p>		
		<table border="1"> <tr> <td>TRUE</td><td> <p>The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle.</p> <p>ENO Output: The output is TRUE and can be set to FALSE separately based on the internal error detection.</p> </td></tr> <tr> <td>FALSE</td><td> <p>The function block or function is not processed by the user program. The following variables and value field assignment are not assigned.</p> <p>The outputs of a function block display the results of the user program cycle that was lastly processed.</p> <p>The outputs of a function display the default values defined for the data type.</p> <p>As long as the function block or function has not been completely processed, the outputs retain the default values defined for the data type.</p> <p>ENO Output: The output is also set to FALSE.</p> </td></tr> </table>	TRUE	<p>The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle.</p> <p>ENO Output: The output is TRUE and can be set to FALSE separately based on the internal error detection.</p>
TRUE	<p>The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle.</p> <p>ENO Output: The output is TRUE and can be set to FALSE separately based on the internal error detection.</p>			
FALSE	<p>The function block or function is not processed by the user program. The following variables and value field assignment are not assigned.</p> <p>The outputs of a function block display the results of the user program cycle that was lastly processed.</p> <p>The outputs of a function display the default values defined for the data type.</p> <p>As long as the function block or function has not been completely processed, the outputs retain the default values defined for the data type.</p> <p>ENO Output: The output is also set to FALSE.</p>			

8.19.2 Outputs

Output	Data type	Description	
OUT	ANY_NUM	Value at IN with changed sign. • If IN is NaN, OUT is NaN as well. • If IN is +INF, OUT is -INF. • If IN is -INF, OUT is +INF.	
ENO	BOOL	TRUE	If EN = TRUE and the function was processed without errors.
		FALSE	<ul style="list-style-type: none"> • If also EN = FALSE or • Error: Data type of IN is USINT, UINT, UDINT or ULINT. • Error: CHS(IN1) exceeds the range of values. • Error: IN is already an exception value for an ANY_REAL (NaN, +INF, -INF).

8.19.3 Special Cases

8.19.3.1 IN as an Unsigned Integer Data Type

Although the value of IN is an unsigned integer (e.g., UINT), IN is interpreted as a signed bit pattern (see also ENO). The signed bit pattern is converted and output as unsigned value.

Depending on the data type, the value of IN is subtracted from the maximum value of the unsigned data type and 1 is added.

The value of OUT is unsigned and is calculated as follows:

$$\text{OUT} = \text{MAX(data type)} - \text{IN} + 1$$

Examples:

Data type	Maximum value	IN	OUT
USINT	255	4	$255 - \text{IN} + 1 = 252$
UINT	65535	4	$65535 - \text{IN} + 1 = 65532$
UDINT	4294967295	4	$4294967295 - \text{IN} + 1 = 4294967292$

8.19.3.2 IN as the Largest Negative Signed Integer:

If IN as the Largest Negative Signed Integer for the data type of IN, no valid positive value exists within the range of values. The unchanged value of IN is output to OUT, and ENO is set to FALSE.

Examples:

Data type	Range of values	OUT
SINT	-128...127	$\text{IN} = -128 \rightarrow \text{OUT} = -128$
INT	-32768...32767	$\text{IN} = -32768 \rightarrow \text{OUT} = -32768$
DINT	-2147483648...2147483647	$\text{IN} = -2147483648 \rightarrow \text{OUT} = -2147483648$

8.20 CHK_REAL

The CHK_REAL function is used to classify REAL values. If the value at the IN input is a normalized or de-normalized ANY_REAL value, all the outputs are FALSE.

8.20.1 Inputs

Input	Data type	Description				
IN	ANY_REAL	Input IN				
EN	BOOL	<p>EN is hidden by default. If EN is connected, it is always TRUE.</p> <p>Show EN/ENO: Right-click a function block or function, and select Show EN/ENO from the context menu.</p> <p>As soon as EN or ENO is connected, it can no longer be hidden.</p> <table border="1"> <tr> <td>TRUE</td><td>The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle. ENO Output: The output is TRUE and can be set to FALSE separately based on the internal error detection.</td></tr> <tr> <td>FALSE</td><td>The function block or function is not processed by the user program. The following variables and value field assignment are not assigned. The outputs of a function block display the results of the user program cycle that was lastly processed. The outputs of a function display the default values defined for the data type. As long as the function block or function has not been completely processed, the outputs retain the default values defined for the data type. ENO Output: The output is also set to FALSE.</td></tr> </table>	TRUE	The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle. ENO Output: The output is TRUE and can be set to FALSE separately based on the internal error detection.	FALSE	The function block or function is not processed by the user program. The following variables and value field assignment are not assigned. The outputs of a function block display the results of the user program cycle that was lastly processed. The outputs of a function display the default values defined for the data type. As long as the function block or function has not been completely processed, the outputs retain the default values defined for the data type. ENO Output: The output is also set to FALSE.
TRUE	The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle. ENO Output: The output is TRUE and can be set to FALSE separately based on the internal error detection.					
FALSE	The function block or function is not processed by the user program. The following variables and value field assignment are not assigned. The outputs of a function block display the results of the user program cycle that was lastly processed. The outputs of a function display the default values defined for the data type. As long as the function block or function has not been completely processed, the outputs retain the default values defined for the data type. ENO Output: The output is also set to FALSE.					

8.20.2 Outputs

Output	Data type	Description	
NAN	BOOL	TRUE	The value of IN is NaN.
		FALSE	The value of IN is not NaN.
+INF	BOOL	TRUE	The value of IN is positive infinite.
		FALSE	The value of IN is not positive infinite.
-INF	BOOL	TRUE	The value of IN is negative infinite.
		FALSE	The value of IN is not negative infinite.
<u>ENO</u>	BOOL	TRUE	If EN is TRUE and the current cycle of the function block or function was processed without errors, ENO is also TRUE. The function cannot result in an error.
		FALSE	Only if EN is also FALSE or while the function block or function is being processed.

9 SELECT Sub-Library

9.1 SEL (Select)

The SEL function enables the choice between two values using a control input.

9.1.1 Inputs

Input	Data type	Description				
G	BOOL	Defines if the value of IN0 or IN1 will be output to OUT.				
IN0	ANY(*)	Input IN0				
IN1	ANY(*)	Input IN1 (*) The inputs must be of the same data type.				
EN	BOOL	<p>EN is hidden by default. If EN is connected, it is always TRUE. Show EN/ENO: Right-click a function block or function, and select Show EN/ENO from the context menu.</p> <p>As soon as EN or ENO is connected, it can no longer be hidden.</p> <table border="1"> <tr> <td>TRUE</td><td> <p>The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle.</p> <p>ENO Output: The output is TRUE and can be set to FALSE separately based on the internal error detection.</p> </td></tr> <tr> <td>FALSE</td><td> <p>The function block or function is not processed by the user program. The following variables and value field assignment are not assigned.</p> <p>The outputs of a function block display the results of the user program cycle that was lastly processed.</p> <p>The outputs of a function display the default values defined for the data type.</p> <p>As long as the function block or function has not been completely processed, the outputs retain the default values defined for the data type.</p> <p>ENO Output: The output is also set to FALSE.</p> </td></tr> </table>	TRUE	<p>The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle.</p> <p>ENO Output: The output is TRUE and can be set to FALSE separately based on the internal error detection.</p>	FALSE	<p>The function block or function is not processed by the user program. The following variables and value field assignment are not assigned.</p> <p>The outputs of a function block display the results of the user program cycle that was lastly processed.</p> <p>The outputs of a function display the default values defined for the data type.</p> <p>As long as the function block or function has not been completely processed, the outputs retain the default values defined for the data type.</p> <p>ENO Output: The output is also set to FALSE.</p>
TRUE	<p>The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle.</p> <p>ENO Output: The output is TRUE and can be set to FALSE separately based on the internal error detection.</p>					
FALSE	<p>The function block or function is not processed by the user program. The following variables and value field assignment are not assigned.</p> <p>The outputs of a function block display the results of the user program cycle that was lastly processed.</p> <p>The outputs of a function display the default values defined for the data type.</p> <p>As long as the function block or function has not been completely processed, the outputs retain the default values defined for the data type.</p> <p>ENO Output: The output is also set to FALSE.</p>					

9.1.2 Outputs

Output	Data type	Description	
OUT	ANY	<ul style="list-style-type: none">• For G = 0: OUT = IN0• For G = 1: OUT = IN1	
<u>ENO</u>	BOOL	TRUE	If EN is TRUE and the current cycle of the function block or function was processed without errors, ENO is also TRUE. The function cannot result in an error.
		FALSE	Only if EN is also FALSE or while the function block or function is being processed.

9.2 MAX (Maximum Value)

The MAX function provides the maximum value of all the used INn inputs to the OUT output.

9.2.1 Inputs

Input	Data type	Description				
IN1	ANY ELEMENTARY	Input IN1				
IN2	ANY ELEMENTARY	Input IN2				
Extendible	ANY ELEMENTARY	Up to IN16				
EN	BOOL	<p>EN is hidden by default. If EN is connected, it is always TRUE. Show EN/ENO: Right-click a function block or function, and select Show EN/ENO from the context menu.</p> <p>As soon as EN or ENO is connected, it can no longer be hidden.</p> <table border="1"> <tr> <td>TRUE</td><td> <p>The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle.</p> <p>ENO Output: The output is TRUE and can be set to FALSE separately based on the internal error detection.</p> </td></tr> <tr> <td>FALSE</td><td> <p>The function block or function is not processed by the user program. The following variables and value field assignment are not assigned.</p> <p>The outputs of a function block display the results of the user program cycle that was lastly processed.</p> <p>The outputs of a function display the default values defined for the data type.</p> <p>As long as the function block or function has not been completely processed, the outputs retain the default values defined for the data type.</p> <p>ENO Output: The output is also set to FALSE.</p> </td></tr> </table>	TRUE	<p>The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle.</p> <p>ENO Output: The output is TRUE and can be set to FALSE separately based on the internal error detection.</p>	FALSE	<p>The function block or function is not processed by the user program. The following variables and value field assignment are not assigned.</p> <p>The outputs of a function block display the results of the user program cycle that was lastly processed.</p> <p>The outputs of a function display the default values defined for the data type.</p> <p>As long as the function block or function has not been completely processed, the outputs retain the default values defined for the data type.</p> <p>ENO Output: The output is also set to FALSE.</p>
TRUE	<p>The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle.</p> <p>ENO Output: The output is TRUE and can be set to FALSE separately based on the internal error detection.</p>					
FALSE	<p>The function block or function is not processed by the user program. The following variables and value field assignment are not assigned.</p> <p>The outputs of a function block display the results of the user program cycle that was lastly processed.</p> <p>The outputs of a function display the default values defined for the data type.</p> <p>As long as the function block or function has not been completely processed, the outputs retain the default values defined for the data type.</p> <p>ENO Output: The output is also set to FALSE.</p>					

9.2.2 Outputs

Output	Data type	Description	
OUT	ANY ELEMENTARY	OUT = The largest value from IN1 to INn  If at least one input of type REAL or LREAL contains the value NaN, the result of OUT is also NaN, and ENO = FALSE.	
<u>ENO</u>	BOOL	TRUE	If EN is TRUE and the current cycle of the function block or function was processed without errors, ENO is also TRUE. The function cannot result in an error.
		FALSE	<ul style="list-style-type: none"> • Only if also EN = FALSE or • Error: At least one input contains the value NaN.

9.3 MIN (Minimum Value)

The MIN function provides the minimum value of all the used INn inputs to the OUT output.

9.3.1 Inputs

Input	Data type	Description				
IN1	ANY ELEMENTARY	Input IN1				
IN2	ANY ELEMENTARY	Input IN2				
Extendible	ANY ELEMENTARY	Up to IN16				
EN	BOOL	<p>EN is hidden by default. If EN is connected, it is always TRUE. Show EN/ENO: Right-click a function block or function, and select Show EN/ENO from the context menu.</p> <p>As soon as EN or ENO is connected, it can no longer be hidden.</p> <table border="1"> <tr> <td>TRUE</td><td> <p>The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle.</p> <p>ENO Output: The output is TRUE and can be set to FALSE separately based on the internal error detection.</p> </td></tr> <tr> <td>FALSE</td><td> <p>The function block or function is not processed by the user program. The following variables and value field assignment are not assigned.</p> <p>The outputs of a function block display the results of the user program cycle that was lastly processed.</p> <p>The outputs of a function display the default values defined for the data type.</p> <p>As long as the function block or function has not been completely processed, the outputs retain the default values defined for the data type.</p> <p>ENO Output: The output is also set to FALSE.</p> </td></tr> </table>	TRUE	<p>The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle.</p> <p>ENO Output: The output is TRUE and can be set to FALSE separately based on the internal error detection.</p>	FALSE	<p>The function block or function is not processed by the user program. The following variables and value field assignment are not assigned.</p> <p>The outputs of a function block display the results of the user program cycle that was lastly processed.</p> <p>The outputs of a function display the default values defined for the data type.</p> <p>As long as the function block or function has not been completely processed, the outputs retain the default values defined for the data type.</p> <p>ENO Output: The output is also set to FALSE.</p>
TRUE	<p>The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle.</p> <p>ENO Output: The output is TRUE and can be set to FALSE separately based on the internal error detection.</p>					
FALSE	<p>The function block or function is not processed by the user program. The following variables and value field assignment are not assigned.</p> <p>The outputs of a function block display the results of the user program cycle that was lastly processed.</p> <p>The outputs of a function display the default values defined for the data type.</p> <p>As long as the function block or function has not been completely processed, the outputs retain the default values defined for the data type.</p> <p>ENO Output: The output is also set to FALSE.</p>					

9.3.2 Outputs

Output	Data type	Description	
OUT	ANY ELEMENTARY	<p>OUT = The lowest value from IN1 to INn</p> <p> If at least one input of type REAL or LREAL contains the value NaN, the result of OUT is also NaN, and ENO = FALSE.</p>	
ENO	BOOL	TRUE	If EN is TRUE and the current cycle of the function block or function was processed without errors, ENO is also TRUE. The function cannot result in an error.
		FALSE	<ul style="list-style-type: none"> • Only if also EN = FALSE or • Error: At least one input contains the value NaN.

9.4 LIMIT (Limiter)

The LIMIT function limits the input value to the range of values defined with MN and MX.

9.4.1 Inputs

Input	Data type	Description				
MN	ANY ELEMENTARY	Lower limit of the range of values				
IN	ANY ELEMENTARY	Value to be limited				
MX	ANY ELEMENTARY	Upper limit of the range of values				
EN	BOOL	<p>EN is hidden by default. If EN is connected, it is always TRUE. Show EN/ENO: Right-click a function block or function, and select Show EN/ENO from the context menu.</p> <p>As soon as EN or ENO is connected, it can no longer be hidden.</p> <table border="1"> <tr> <td>TRUE</td><td> <p>The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle.</p> <p>ENO Output: The output is TRUE and can be set to FALSE separately based on the internal error detection.</p> </td></tr> <tr> <td>FALSE</td><td> <p>The function block or function is not processed by the user program. The following variables and value field assignment are not assigned.</p> <p>The outputs of a function block display the results of the user program cycle that was lastly processed.</p> <p>The outputs of a function display the default values defined for the data type.</p> <p>As long as the function block or function has not been completely processed, the outputs retain the default values defined for the data type.</p> <p>ENO Output: The output is also set to FALSE.</p> </td></tr> </table>	TRUE	<p>The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle.</p> <p>ENO Output: The output is TRUE and can be set to FALSE separately based on the internal error detection.</p>	FALSE	<p>The function block or function is not processed by the user program. The following variables and value field assignment are not assigned.</p> <p>The outputs of a function block display the results of the user program cycle that was lastly processed.</p> <p>The outputs of a function display the default values defined for the data type.</p> <p>As long as the function block or function has not been completely processed, the outputs retain the default values defined for the data type.</p> <p>ENO Output: The output is also set to FALSE.</p>
TRUE	<p>The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle.</p> <p>ENO Output: The output is TRUE and can be set to FALSE separately based on the internal error detection.</p>					
FALSE	<p>The function block or function is not processed by the user program. The following variables and value field assignment are not assigned.</p> <p>The outputs of a function block display the results of the user program cycle that was lastly processed.</p> <p>The outputs of a function display the default values defined for the data type.</p> <p>As long as the function block or function has not been completely processed, the outputs retain the default values defined for the data type.</p> <p>ENO Output: The output is also set to FALSE.</p>					

9.4.2 Outputs

Output	Data type	Description	
OUT	ANY ELEMENTARY	IN is limited to the range of values of MN and MX.	
		IN > MX:	OUT = MX
		IN < MN:	OUT = MN
ENO	BOOL	MN ≤ IN ≤ MX:	OUT = IN
		TRUE	If EN is TRUE and the current cycle of the function block or function was processed without errors, ENO is also TRUE. The function cannot result in an error.
		FALSE	Only if EN is also FALSE or while the function block or function is being processed.

9.4.3 Exceptions for REAL/LREAL Type Conversion

NaN input values do not affect ENO.

IN = NaN	OUT = NaN
MN = NaN	Behaves like MN = -INF, i.e., there is no lower limit.
MX = NaN	Behaves like MN = +INF, i.e., there is no upper limit.

9.4.4 Implementation in Structured Text (ST)

```
OUT := MIN( MAX( IN, MN ), MX )
```

9.5 MUX (Multiplexer)

The MUX function displays the value of the IN(K) input to OUT, where $0 \leq K < n$.

9.5.1 Inputs

Input	Data type	Description				
K	ANY_INT	<p>Used to select the multiplexer input with the value that should be output to OUT.</p> <p>Range of values: $0 \leq K < n$ where $n =$ number of multiplexer inputs</p>				
IN0	ANY(*)	Multiplexer input for $K = 0$				
IN1	ANY(*)	Multiplexer input for $K = 1$				
Extendible	ANY(*)	<p>A maximum of 16 multiplexer inputs (*) The multiplexer inputs must be of the same data type.</p>				
<u>EN</u>	BOOL	<p>EN is hidden by default. If EN is connected, it is always TRUE.</p> <p>Show EN/ENO: Right-click a function block or function, and select Show EN/ENO from the context menu.</p> <p>As soon as EN or ENO is connected, it can no longer be hidden.</p> <table border="1"> <tr> <td>TRUE</td><td> <p>The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle.</p> <p>ENO Output: The output is TRUE and can be set to FALSE separately based on the internal error detection.</p> </td></tr> <tr> <td>FALSE</td><td> <p>The function block or function is not processed by the user program. The following variables and value field assignment are not assigned.</p> <p>The outputs of a function block display the results of the user program cycle that was lastly processed.</p> <p>The outputs of a function display the default values defined for the data type.</p> <p>As long as the function block or function has not been completely processed, the outputs retain the default values defined for the data type.</p> <p>ENO Output: The output is also set to FALSE.</p> </td></tr> </table>	TRUE	<p>The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle.</p> <p>ENO Output: The output is TRUE and can be set to FALSE separately based on the internal error detection.</p>	FALSE	<p>The function block or function is not processed by the user program. The following variables and value field assignment are not assigned.</p> <p>The outputs of a function block display the results of the user program cycle that was lastly processed.</p> <p>The outputs of a function display the default values defined for the data type.</p> <p>As long as the function block or function has not been completely processed, the outputs retain the default values defined for the data type.</p> <p>ENO Output: The output is also set to FALSE.</p>
TRUE	<p>The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle.</p> <p>ENO Output: The output is TRUE and can be set to FALSE separately based on the internal error detection.</p>					
FALSE	<p>The function block or function is not processed by the user program. The following variables and value field assignment are not assigned.</p> <p>The outputs of a function block display the results of the user program cycle that was lastly processed.</p> <p>The outputs of a function display the default values defined for the data type.</p> <p>As long as the function block or function has not been completely processed, the outputs retain the default values defined for the data type.</p> <p>ENO Output: The output is also set to FALSE.</p>					

9.5.2 Outputs

Output	Data type	Description	
OUT	ANY	K = 0	OUT = IN0
		K = 1	OUT = IN1
		K = 2	OUT = IN2
		K = x	OUT = INx
		For value of K that are out of the valid range of values: <ul style="list-style-type: none"> • K < 0 → OUT = IN0 • K > n-1 → OUT = IN(n-1) where n = number of multiplexer inputs	
<u>ENO</u>	BOOL	TRUE	If EN = TRUE and the function was processed without errors.
		FALSE	<ul style="list-style-type: none"> • If also EN = FALSE or • Error: The value of K is out of the valid range of values.

9.5.3 Implementation in Structured Text (ST)

```

IF (k >= 0) AND (K < n)
  THEN OUT := IN(k);
ELIF ( k< 0 )
  THEN OUT := IN0;
ELIF ( k >= n )
  THEN OUT := INn-1;
ENDIF
  
```

10 TIME Sub-Library

10.1 DIV_TIME

The DIV_TIME function calculates the quotient of a time and a numerical value.

10.1.1 Inputs

Input	Data type	Description				
IN1	TIME	Input IN1				
IN2	ANY_NUM	Input IN2				
EN	BOOL	<p>EN is hidden by default. If EN is connected, it is always TRUE. Show EN/ENO: Right-click a function block or function, and select Show EN/ENO from the context menu.</p> <p>As soon as EN or ENO is connected, it can no longer be hidden.</p> <table border="1"> <tr> <td>TRUE</td><td> The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle. ENO Output: The output is TRUE and can be set to FALSE separately based on the internal error detection. </td></tr> <tr> <td>FALSE</td><td> The function block or function is not processed by the user program. The following variables and value field assignment are not assigned. The outputs of a function block display the results of the user program cycle that was lastly processed. The outputs of a function display the default values defined for the data type. As long as the function block or function has not been completely processed, the outputs retain the default values defined for the data type. ENO Output: The output is also set to FALSE. </td></tr> </table>	TRUE	The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle. ENO Output: The output is TRUE and can be set to FALSE separately based on the internal error detection.	FALSE	The function block or function is not processed by the user program. The following variables and value field assignment are not assigned. The outputs of a function block display the results of the user program cycle that was lastly processed. The outputs of a function display the default values defined for the data type. As long as the function block or function has not been completely processed, the outputs retain the default values defined for the data type. ENO Output: The output is also set to FALSE.
TRUE	The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle. ENO Output: The output is TRUE and can be set to FALSE separately based on the internal error detection.					
FALSE	The function block or function is not processed by the user program. The following variables and value field assignment are not assigned. The outputs of a function block display the results of the user program cycle that was lastly processed. The outputs of a function display the default values defined for the data type. As long as the function block or function has not been completely processed, the outputs retain the default values defined for the data type. ENO Output: The output is also set to FALSE.					

10.1.2 Outputs

Output	Data type	Description				
OUT	TIME	<p>OUT = IN1 / IN2.</p> <p>i If the data type of IN2 is ANY_REAL, the data type of IN changes to LREAL for determining the TIME data type. Due to this, the accuracy of the results is reduced from 64 bits to 53 bits. This also leads to rounding errors if large values (greater than 104 million days) are used in the calculations.</p>				
<u>ENO</u>	BOOL	<table border="1"> <tr> <td>TRUE</td><td> <ul style="list-style-type: none"> • EN = TRUE and • The mathematical function was performed without errors. </td></tr> <tr> <td>FALSE</td><td> <ul style="list-style-type: none"> • EN = FALSE or • Error: The mathematical function was performed with error (overflow or division by 0). </td></tr> </table>	TRUE	<ul style="list-style-type: none"> • EN = TRUE and • The mathematical function was performed without errors. 	FALSE	<ul style="list-style-type: none"> • EN = FALSE or • Error: The mathematical function was performed with error (overflow or division by 0).
TRUE	<ul style="list-style-type: none"> • EN = TRUE and • The mathematical function was performed without errors. 					
FALSE	<ul style="list-style-type: none"> • EN = FALSE or • Error: The mathematical function was performed with error (overflow or division by 0). 					

10.2 MUL_TIME

The MUL_TIME function calculates the product of a time and a numerical value.

10.2.1 Inputs

Input	Data type	Description				
IN1	TIME	Input IN1				
IN2	ANY_NUM	Input IN2				
EN	BOOL	<p>EN is hidden by default. If EN is connected, it is always TRUE.</p> <p>Show EN/ENO: Right-click a function block or function, and select Show EN/ENO from the context menu.</p> <p>As soon as EN or ENO is connected, it can no longer be hidden.</p> <table border="1"> <tr> <td>TRUE</td><td> <p>The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle.</p> <p>ENO Output: The output is TRUE and can be set to FALSE separately based on the internal error detection.</p> </td></tr> <tr> <td>FALSE</td><td> <p>The function block or function is not processed by the user program. The following variables and value field assignment are not assigned.</p> <p>The outputs of a function block display the results of the user program cycle that was lastly processed.</p> <p>The outputs of a function display the default values defined for the data type.</p> <p>As long as the function block or function has not been completely processed, the outputs retain the default values defined for the data type.</p> <p>ENO Output: The output is also set to FALSE.</p> </td></tr> </table>	TRUE	<p>The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle.</p> <p>ENO Output: The output is TRUE and can be set to FALSE separately based on the internal error detection.</p>	FALSE	<p>The function block or function is not processed by the user program. The following variables and value field assignment are not assigned.</p> <p>The outputs of a function block display the results of the user program cycle that was lastly processed.</p> <p>The outputs of a function display the default values defined for the data type.</p> <p>As long as the function block or function has not been completely processed, the outputs retain the default values defined for the data type.</p> <p>ENO Output: The output is also set to FALSE.</p>
TRUE	<p>The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle.</p> <p>ENO Output: The output is TRUE and can be set to FALSE separately based on the internal error detection.</p>					
FALSE	<p>The function block or function is not processed by the user program. The following variables and value field assignment are not assigned.</p> <p>The outputs of a function block display the results of the user program cycle that was lastly processed.</p> <p>The outputs of a function display the default values defined for the data type.</p> <p>As long as the function block or function has not been completely processed, the outputs retain the default values defined for the data type.</p> <p>ENO Output: The output is also set to FALSE.</p>					

10.2.2 Outputs

Output	Data type	Description				
OUT	TIME	<p>OUT = IN1 * IN2</p> <p>i If the data type of IN2 is ANY_REAL, the data type of IN changes to LREAL for determining the TIME data type. Due to this, the accuracy of the results is reduced from 64 bits to 53 bits. This also leads to rounding errors if large values (greater than 104 million days) are used in the calculations.</p>				
<u>ENO</u>	BOOL	<table border="1"> <tr> <td>TRUE</td><td> <ul style="list-style-type: none"> EN = TRUE and The mathematical function was performed without errors. </td></tr> <tr> <td>FALSE</td><td> <ul style="list-style-type: none"> EN = FALSE or Error: Result overflow. </td></tr> </table>	TRUE	<ul style="list-style-type: none"> EN = TRUE and The mathematical function was performed without errors. 	FALSE	<ul style="list-style-type: none"> EN = FALSE or Error: Result overflow.
TRUE	<ul style="list-style-type: none"> EN = TRUE and The mathematical function was performed without errors. 					
FALSE	<ul style="list-style-type: none"> EN = FALSE or Error: Result overflow. 					

10.3 ADD_TIME

The ADD_TIME function calculates the sum of two time values.

10.3.1 Inputs

Input	Data type	Description				
IN1	TIME	Input IN1				
IN2	TIME	Input IN2				
EN	BOOL	<p>EN is hidden by default. If EN is connected, it is always TRUE.</p> <p>Show EN/ENO: Right-click a function block or function, and select Show EN/ENO from the context menu.</p> <p>As soon as EN or ENO is connected, it can no longer be hidden.</p> <table border="1"> <tr> <td>TRUE</td><td> <p>The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle.</p> <p>ENO Output: The output is TRUE and can be set to FALSE separately based on the internal error detection.</p> </td></tr> <tr> <td>FALSE</td><td> <p>The function block or function is not processed by the user program. The following variables and value field assignment are not assigned.</p> <p>The outputs of a function block display the results of the user program cycle that was lastly processed.</p> <p>The outputs of a function display the default values defined for the data type.</p> <p>As long as the function block or function has not been completely processed, the outputs retain the default values defined for the data type.</p> <p>ENO Output: The output is also set to FALSE.</p> </td></tr> </table>	TRUE	<p>The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle.</p> <p>ENO Output: The output is TRUE and can be set to FALSE separately based on the internal error detection.</p>	FALSE	<p>The function block or function is not processed by the user program. The following variables and value field assignment are not assigned.</p> <p>The outputs of a function block display the results of the user program cycle that was lastly processed.</p> <p>The outputs of a function display the default values defined for the data type.</p> <p>As long as the function block or function has not been completely processed, the outputs retain the default values defined for the data type.</p> <p>ENO Output: The output is also set to FALSE.</p>
TRUE	<p>The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle.</p> <p>ENO Output: The output is TRUE and can be set to FALSE separately based on the internal error detection.</p>					
FALSE	<p>The function block or function is not processed by the user program. The following variables and value field assignment are not assigned.</p> <p>The outputs of a function block display the results of the user program cycle that was lastly processed.</p> <p>The outputs of a function display the default values defined for the data type.</p> <p>As long as the function block or function has not been completely processed, the outputs retain the default values defined for the data type.</p> <p>ENO Output: The output is also set to FALSE.</p>					

10.3.2 Outputs

Output	Data type	Description										
OUT	TIME	<p>$OUT = IN1 + IN2$</p> <p>If the range limits for OUT are exceeded, the overflow is signaled on ENO and output to OUT 1.</p> <p>Examples:</p> <table border="1"> <thead> <tr> <th>Data type</th><th>Function</th><th>OUT</th></tr> </thead> <tbody> <tr> <td rowspan="3">TIME</td><td>ADD_TIME(T#60s; T#60s)</td><td>T#2m</td></tr> <tr> <td>ADD_TIME(T#-1h; T#-1h)</td><td>T#-2h</td></tr> <tr> <td>ADD_TIME (54000000000d; 54000000000d)</td><td>T#-105503982334d... (Overflow)</td></tr> </tbody> </table>	Data type	Function	OUT	TIME	ADD_TIME(T#60s; T#60s)	T#2m	ADD_TIME(T#-1h; T#-1h)	T#-2h	ADD_TIME (54000000000d; 54000000000d)	T#-105503982334d... (Overflow)
Data type	Function	OUT										
TIME	ADD_TIME(T#60s; T#60s)	T#2m										
	ADD_TIME(T#-1h; T#-1h)	T#-2h										
	ADD_TIME (54000000000d; 54000000000d)	T#-105503982334d... (Overflow)										
<u>ENO</u>	BOOL	<table border="1"> <tr> <td>TRUE</td><td>If EN = TRUE and the function was processed without errors.</td></tr> <tr> <td>FALSE</td><td> <ul style="list-style-type: none"> If also EN = FALSE or Error: The value of OUT1 is exceeded. The overflow value is output to OUT1. </td></tr> </table>	TRUE	If EN = TRUE and the function was processed without errors.	FALSE	<ul style="list-style-type: none"> If also EN = FALSE or Error: The value of OUT1 is exceeded. The overflow value is output to OUT1. 						
TRUE	If EN = TRUE and the function was processed without errors.											
FALSE	<ul style="list-style-type: none"> If also EN = FALSE or Error: The value of OUT1 is exceeded. The overflow value is output to OUT1. 											

10.4 SUB_TIME

The SUB_TIME function calculates the difference between two time values.

10.4.1 Inputs

Input	Data type	Description				
IN1	TIME	Input IN1				
IN2	TIME	Input IN2				
EN	BOOL	<p>EN is hidden by default. If EN is connected, it is always TRUE. Show EN/ENO: Right-click a function block or function, and select Show EN/ENO from the context menu.</p> <p>As soon as EN or ENO is connected, it can no longer be hidden.</p> <table border="1"> <tr> <td>TRUE</td><td> <p>The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle.</p> <p>ENO Output: The output is TRUE and can be set to FALSE separately based on the internal error detection.</p> </td></tr> <tr> <td>FALSE</td><td> <p>The function block or function is not processed by the user program. The following variables and value field assignment are not assigned.</p> <p>The outputs of a function block display the results of the user program cycle that was lastly processed.</p> <p>The outputs of a function display the default values defined for the data type.</p> <p>As long as the function block or function has not been completely processed, the outputs retain the default values defined for the data type.</p> <p>ENO Output: The output is also set to FALSE.</p> </td></tr> </table>	TRUE	<p>The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle.</p> <p>ENO Output: The output is TRUE and can be set to FALSE separately based on the internal error detection.</p>	FALSE	<p>The function block or function is not processed by the user program. The following variables and value field assignment are not assigned.</p> <p>The outputs of a function block display the results of the user program cycle that was lastly processed.</p> <p>The outputs of a function display the default values defined for the data type.</p> <p>As long as the function block or function has not been completely processed, the outputs retain the default values defined for the data type.</p> <p>ENO Output: The output is also set to FALSE.</p>
TRUE	<p>The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle.</p> <p>ENO Output: The output is TRUE and can be set to FALSE separately based on the internal error detection.</p>					
FALSE	<p>The function block or function is not processed by the user program. The following variables and value field assignment are not assigned.</p> <p>The outputs of a function block display the results of the user program cycle that was lastly processed.</p> <p>The outputs of a function display the default values defined for the data type.</p> <p>As long as the function block or function has not been completely processed, the outputs retain the default values defined for the data type.</p> <p>ENO Output: The output is also set to FALSE.</p>					

10.4.2 Outputs

Output	Data type	Description				
OUT	TIME	OUT = IN1 - IN 2				
ENO	BOOL	<table border="1"> <tr> <td>TRUE</td><td>If EN = TRUE and the function was processed without errors.</td></tr> <tr> <td>FALSE</td><td> <ul style="list-style-type: none"> If also EN = FALSE or Error: The range of values of OUT is exceeded </td></tr> </table>	TRUE	If EN = TRUE and the function was processed without errors.	FALSE	<ul style="list-style-type: none"> If also EN = FALSE or Error: The range of values of OUT is exceeded
TRUE	If EN = TRUE and the function was processed without errors.					
FALSE	<ul style="list-style-type: none"> If also EN = FALSE or Error: The range of values of OUT is exceeded 					

11 TIMER Sub-Library

11.1 General Timer Behavior

A timer's expiration time indicates how much time has elapsed since the timer started.

All the timer of a user program use a consistent time basis (system tick) to determine their expiration time. Each user program has its specific system tick. The system tick is refreshed at the beginning of each program cycle and remains constant during the program cycle. This means that the timer resolution depends on the duration of the user program cycle.

Since the system tick is constant during one program cycle, a timer's expiration time does not change. If several timers are started within a program cycle, regardless of whether at the beginning or end of the program cycle, all timers record the same expiration time.

If a running timer is temporarily switched off through the EN input, it nevertheless notices the elapsed time since the system tick continues to run.

11.2 Behavior of Timers with Retain Feature

In the following sections, timer instances with retain feature are referred to as retain timers.

The system tick provided by the controller behaves in an undefined manner during a warm start, so that the timer function blocks TON, TOF and TP do not always operate precisely in combination with the retain feature.

In the first program cycle after a warm start, a running retain timer does not detect system tick changes. Displays the same expiration time shown in the last complete program cycle prior to the warm start. This means that the duration of the last program cycle prior to the warm start is not recorded. The precision of the retain timer decreases with each warm start.

Due to the impaired precision of retain timers, they should only be used if the timer runtime is significantly higher than the cycle time of the user program.

The behavior of retain timers depends on the *Minimum Configuration Version* configured in the resource properties.

- [Minimum Configuration Version up to V7](#): Retain timers have an undefined warm start behavior.
- [Minimum Configuration Version V7 and higher](#): Retain timers only measure the time in which the user program was in the RUN state (and in RUN sub-states). Times outside the RUN state are not recorded.

11.3 TOF

The TOF function bloc is used to delay the falling edge (time off delayer).

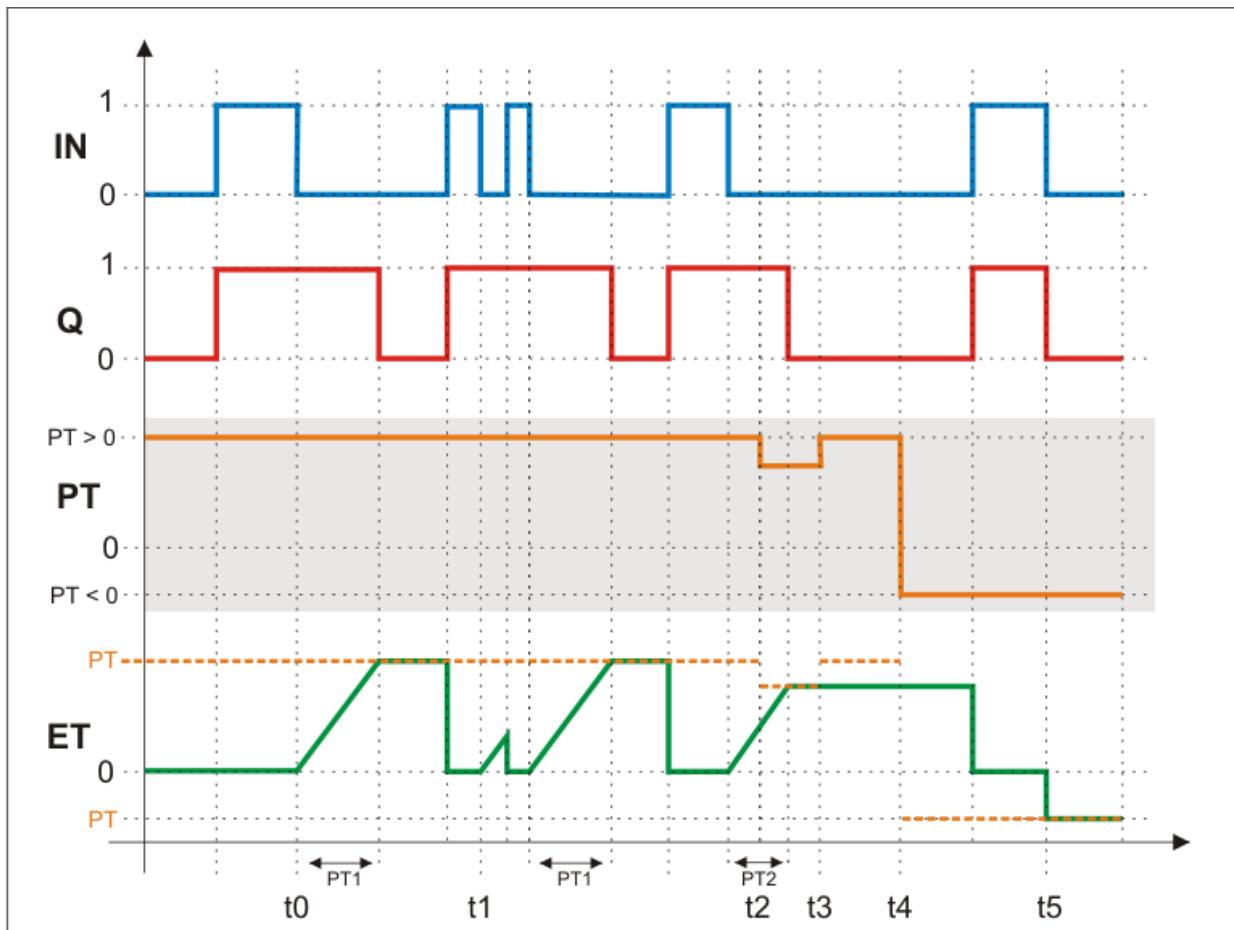
11.3.1 Inputs

Input	Data type	Description				
IN	BOOL	A falling edge at the IN input is output to Q with a delay equal to the time defined at the PT input. To this end, the pulse pause value must be greater than PT.				
PT	TIME	Delay until Q = FALSE. Changes to PT do not always have an immediate effect (see graphic below).				
EN	BOOL	<p>EN is hidden by default. If EN is connected, it is always TRUE.</p> <p>Show EN/ENO: Right-click a function block or function, and select Show EN/ENO from the context menu.</p> <p>As soon as EN or ENO is connected, it can no longer be hidden.</p> <table border="1"> <tr> <td>TRUE</td><td> The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle. ENO Output: The output is TRUE and can be set to FALSE separately based on the internal error detection. </td></tr> <tr> <td>FALSE</td><td> The function block or function is not processed by the user program. The following variables and value field assignment are not assigned. The outputs of a function block display the results of the user program cycle that was lastly processed. The outputs of a function display the default values defined for the data type. As long as the function block or function has not been completely processed, the outputs retain the default values defined for the data type. ENO Output: The output is also set to FALSE. </td></tr> </table>	TRUE	The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle. ENO Output: The output is TRUE and can be set to FALSE separately based on the internal error detection.	FALSE	The function block or function is not processed by the user program. The following variables and value field assignment are not assigned. The outputs of a function block display the results of the user program cycle that was lastly processed. The outputs of a function display the default values defined for the data type. As long as the function block or function has not been completely processed, the outputs retain the default values defined for the data type. ENO Output: The output is also set to FALSE.
TRUE	The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle. ENO Output: The output is TRUE and can be set to FALSE separately based on the internal error detection.					
FALSE	The function block or function is not processed by the user program. The following variables and value field assignment are not assigned. The outputs of a function block display the results of the user program cycle that was lastly processed. The outputs of a function display the default values defined for the data type. As long as the function block or function has not been completely processed, the outputs retain the default values defined for the data type. ENO Output: The output is also set to FALSE.					

11.3.2 Outputs

Output	Data type	Description	
Q	BOOL	TRUE	A rising edge at IN immediately sets Q to TRUE.
		FALSE	A falling edge at IN sets Q to FALSE (Q=FALSE) after a delay of PT .
ET	TIME	Time elapsed since falling edge at IN. If ET = PT, then Q = FALSE.	
<u>ENO</u>	BOOL	TRUE	If EN is TRUE and the current cycle of the function block or function was processed without errors, ENO is also TRUE. The function cannot result in an error.
		FALSE	Only if EN is also FALSE or while the function block or function is being processed.

11.3.3 Graphic on TOF Behavior



t_0	The falling edge has a delay equal to $PT1$.
t_1	Q is not set to FALSE since the value of the pulse pause at IN is less than PT .
t_2	While the timer is running, PT changes (in the example: reduced from $PT1$ to $PT2$). The change affects the current falling edge.
t_3	PT changes after the timer is out (in the example: increased from $PT2$ to $PT1$). ET and Q retain their values. The change only affects the next falling edge at IN .
t_4	PT changes after the timer is out (in the example: reduced to minus). ET and Q retain their values. The change only affects the next falling edge at IN .
t_5	When the falling edge is on IN , $PT \leq 0$. Immediately after $IN = \text{FALSE}$, $ET = PT$ (negative values are also possible!) and $Q = \text{FALSE}$.

11.4 TON

The TON function bloc is used to delay the rising edge (time on delayer).

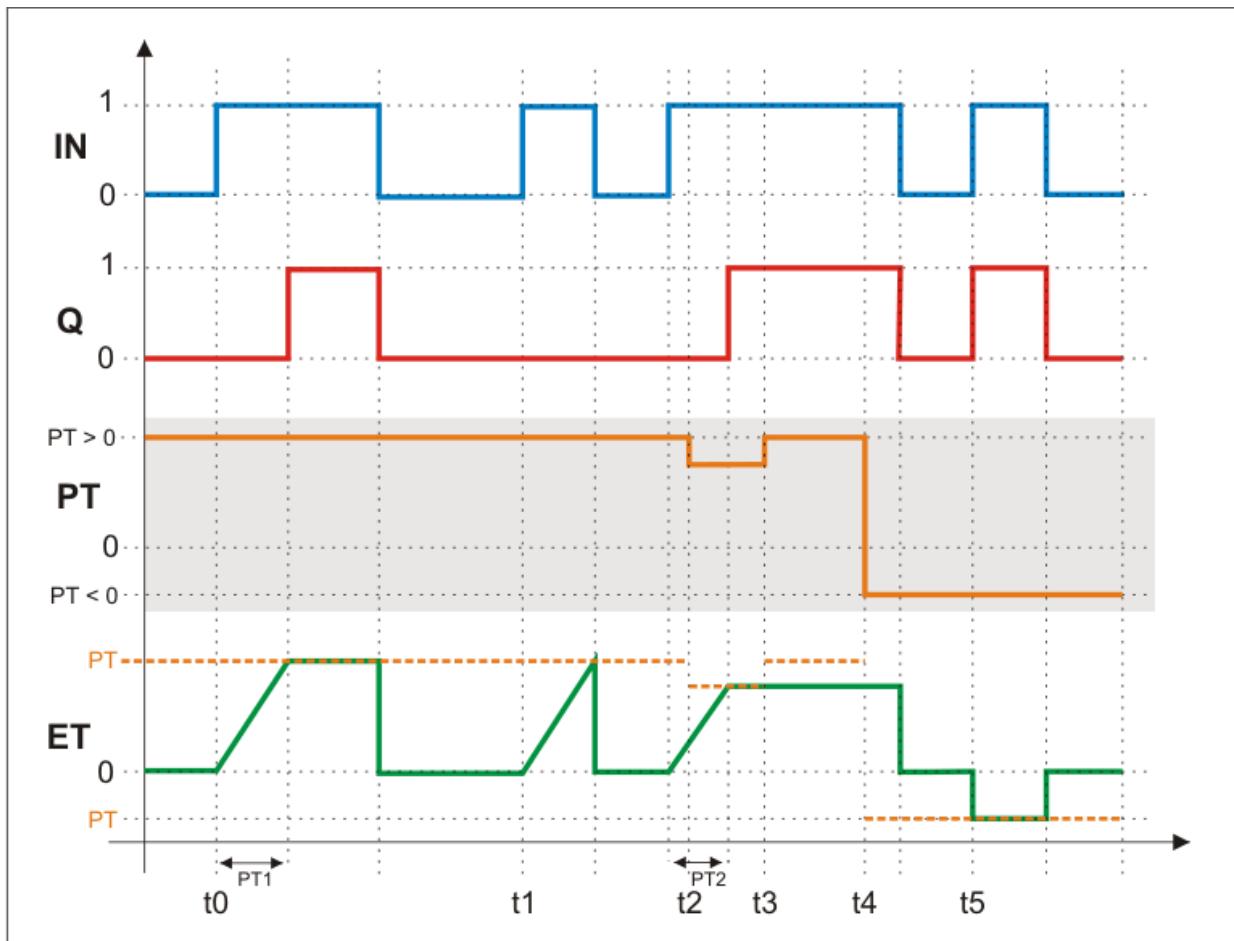
11.4.1 Inputs

Input	Data type	Description				
IN	BOOL	A rising edge at the IN input is output to Q with a delay equal to the time defined at the PT input. To this end, the pulse value must be greater than PT.				
PT	TIME	Delay until Q = TRUE. Changes to PT do not always have an immediate effect (see graphic below).				
EN	BOOL	<p>EN is hidden by default. If EN is connected, it is always TRUE.</p> <p>Show EN/ENO: Right-click a function block or function, and select Show EN/ENO from the context menu.</p> <p>As soon as EN or ENO is connected, it can no longer be hidden.</p> <table border="1"> <tr> <td>TRUE</td><td> <p>The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle.</p> <p>ENO Output: The output is TRUE and can be set to FALSE separately based on the internal error detection.</p> </td></tr> <tr> <td>FALSE</td><td> <p>The function block or function is not processed by the user program. The following variables and value field assignment are not assigned.</p> <p>The outputs of a function block display the results of the user program cycle that was lastly processed.</p> <p>The outputs of a function display the default values defined for the data type.</p> <p>As long as the function block or function has not been completely processed, the outputs retain the default values defined for the data type.</p> <p>ENO Output: The output is also set to FALSE.</p> </td></tr> </table>	TRUE	<p>The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle.</p> <p>ENO Output: The output is TRUE and can be set to FALSE separately based on the internal error detection.</p>	FALSE	<p>The function block or function is not processed by the user program. The following variables and value field assignment are not assigned.</p> <p>The outputs of a function block display the results of the user program cycle that was lastly processed.</p> <p>The outputs of a function display the default values defined for the data type.</p> <p>As long as the function block or function has not been completely processed, the outputs retain the default values defined for the data type.</p> <p>ENO Output: The output is also set to FALSE.</p>
TRUE	<p>The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle.</p> <p>ENO Output: The output is TRUE and can be set to FALSE separately based on the internal error detection.</p>					
FALSE	<p>The function block or function is not processed by the user program. The following variables and value field assignment are not assigned.</p> <p>The outputs of a function block display the results of the user program cycle that was lastly processed.</p> <p>The outputs of a function display the default values defined for the data type.</p> <p>As long as the function block or function has not been completely processed, the outputs retain the default values defined for the data type.</p> <p>ENO Output: The output is also set to FALSE.</p>					

11.4.2 Outputs

Output	Data type	Description	
Q	BOOL	TRUE	A rising edge at IN sets Q1 to TRUE with a delay of PT .
		FALSE	A falling edge on IN immediately sets Q to FALSE (Q=FALSE).
ET	TIME	Time elapsed since rising edge at IN (see graphic below).	
ENO	BOOL	TRUE	If EN is TRUE and the current cycle of the function block or function was processed without errors, ENO is also TRUE. The function cannot result in an error.
		FALSE	Only if EN is also FALSE or while the function block or function is being processed.

11.4.3 Graphic on TON Behavior



t0	The rising edge has a delay equal to PT_1 .
t1	Q is not set to TRUE since the value of the pulse at IN is less than PT.
t2	While the timer is running, PT changes (in the example: reduced from PT_1 to PT_2). The change affects the current rising edge.
t3	PT changes after the timer is out (in the example: increased from PT_2 to PT_1). ET and Q retain their values. The change only affects the next rising edge at IN.
t4	PT changes after the timer is out (in the example: reduced to minus). ET and Q retain their values. The change only affects the next rising edge at IN.
t5	When the rising edge is at IN, PT is ≤ 0 . Immediately after IN = TRUE, ET = PT (negative values are also possible!) and Q = TRUE.

11.5 TP

The TP function block is used as pulse generator with adjustable pulse duration.

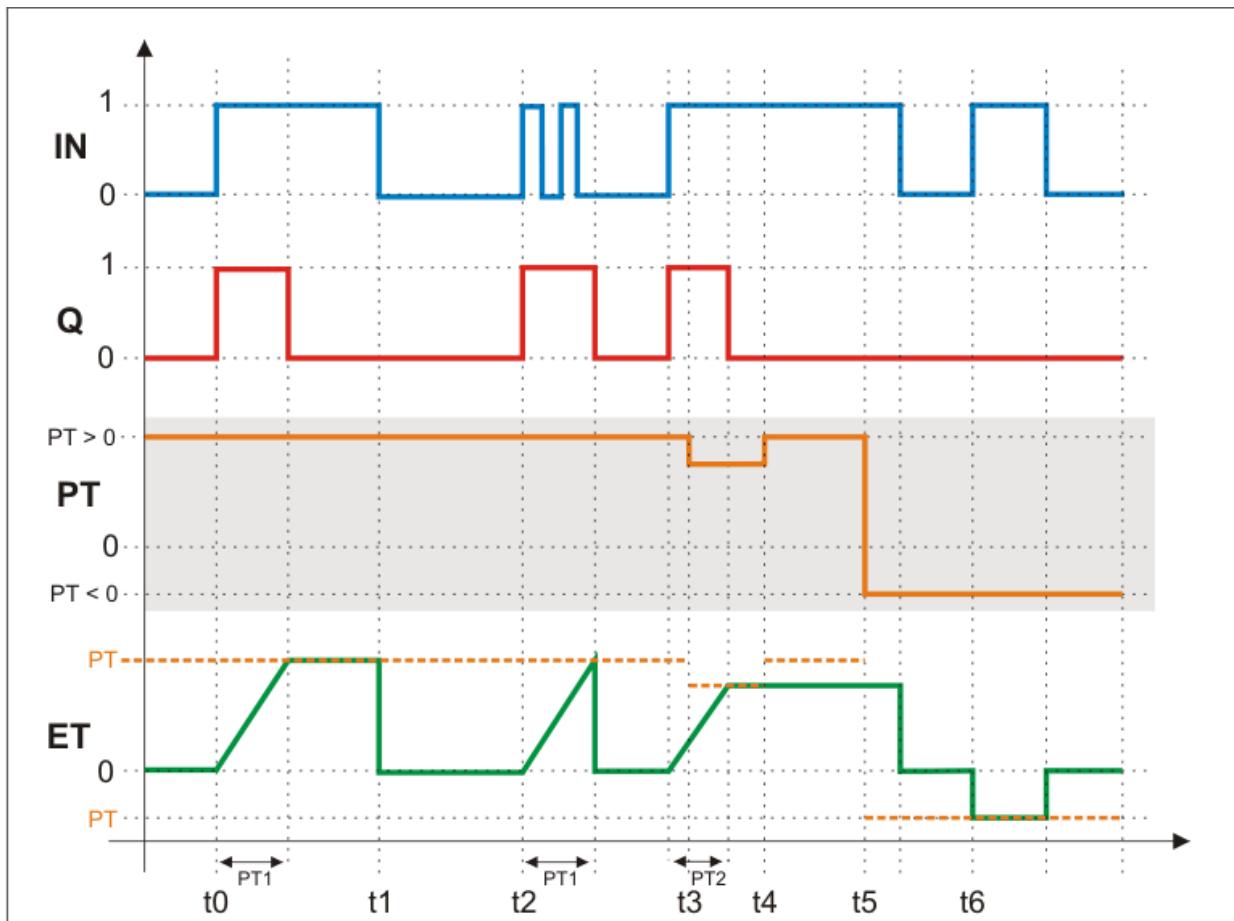
11.5.1 Inputs

Input	Data type	Description				
IN	BOOL	A rising edge at the IN input immediately sets the Q output to TRUE. A falling edge at the IN input has no effect on the Q output.				
PT	TIME	Defines how long Q = TRUE. Changes to PT do not always have an immediate effect (see graphic below).				
<u>EN</u>	BOOL	<p>EN is hidden by default. If EN is connected, it is always TRUE. Show EN/ENO: Right-click a function block or function, and select Show EN/ENO from the context menu.</p> <p>As soon as EN or ENO is connected, it can no longer be hidden.</p> <table border="1"> <tr> <td>TRUE</td><td> <p>The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle.</p> <p>ENO Output: The output is TRUE and can be set to FALSE separately based on the internal error detection.</p> </td></tr> <tr> <td>FALSE</td><td> <p>The function block or function is not processed by the user program. The following variables and value field assignment are not assigned.</p> <p>The outputs of a function block display the results of the user program cycle that was lastly processed.</p> <p>The outputs of a function display the default values defined for the data type.</p> <p>As long as the function block or function has not been completely processed, the outputs retain the default values defined for the data type.</p> <p>ENO Output: The output is also set to FALSE.</p> </td></tr> </table>	TRUE	<p>The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle.</p> <p>ENO Output: The output is TRUE and can be set to FALSE separately based on the internal error detection.</p>	FALSE	<p>The function block or function is not processed by the user program. The following variables and value field assignment are not assigned.</p> <p>The outputs of a function block display the results of the user program cycle that was lastly processed.</p> <p>The outputs of a function display the default values defined for the data type.</p> <p>As long as the function block or function has not been completely processed, the outputs retain the default values defined for the data type.</p> <p>ENO Output: The output is also set to FALSE.</p>
TRUE	<p>The function block or function is processed by the user program. The outputs of the function block or function contain the results of the current user program cycle.</p> <p>ENO Output: The output is TRUE and can be set to FALSE separately based on the internal error detection.</p>					
FALSE	<p>The function block or function is not processed by the user program. The following variables and value field assignment are not assigned.</p> <p>The outputs of a function block display the results of the user program cycle that was lastly processed.</p> <p>The outputs of a function display the default values defined for the data type.</p> <p>As long as the function block or function has not been completely processed, the outputs retain the default values defined for the data type.</p> <p>ENO Output: The output is also set to FALSE.</p>					

11.5.2 Outputs

Output	Data type	Description	
Q	BOOL	TRUE	A positive edge at IN immediately sets Q to TRUE for the duration of PT.
		FALSE	After PT has expired, Q is set to FALSE, irrespective of the value of IN input.
ET	TIME	Time elapsed since the rising edge at IN.	
		If IN = TRUE and ET = PT	ET = PT
<u>ENO</u>	BOOL	TRUE	If EN is TRUE and the current cycle of the function block or function was processed without errors, ENO is also TRUE. The function cannot result in an error.
		FALSE	Only if EN is also FALSE or while the function block or function is being processed.

11.5.3 Graphic on TP Behavior



t0	With the rising edge at IN, Q is set to TRUE. After PT1 has expired, Q is set to FALSE.
t1	The falling edge at IN resets ET.
t2	PT1 defines how long Q = TRUE. If the pulse value at IN is less than PT1, the timed out timer is reset immediately.
t3	While the timer is running, PT changes (in the example: reduced from PT1 to PT2). The change affects the current pulse value at Q.
t4	PT changes after the timer is out (in the example: increased from PT2 to PT1). ET and Q retain their values. The change only affects the next rising edge at IN.
t5	PT changes after the timer is out (in the example: reduced to minus). ET and Q retain their values. The change only affects the next rising edge at IN.
t6	When the rising edge is at IN, PT is ≤ 0 . Immediately after IN = TRUE, ET = PT (negative values are also possible!) and Q remains FALSE since the pulse duration is 0.

12 Sequential Function Charts (SFC)

Like FBD, SFC is also a graphical programming language in accordance with IEC 61131-3. Sequential function charts are a further development of step sequence programming. The sequential function chart language is used to divide complex tasks into smaller units and define the control flow between these units. In particular, this programming language is suitable for tasks in which the individual functions are integrated into an overall process.

A SFC program is composed of networks made up of steps (→ Step), actions (→ Action) and transitions (→ Transition).

In a SFC network, only one step can be active at a time. A step enabling condition controls when the transition occurs. Only when the step enabling condition is met, new steps can be branched. The steps can be branched sequentially, in parallel or alternatively.

SFC programming is defined in the FBD Editor (Function Block Diagram Editor). The following four **SFC Elements** are available in the **Blocks** tab of the Object Panel:

- Initial Step
- Step
- Transition
- Action

12.1 Working with Step Sequences

Use the mouse to connect steps, transitions and actions in the drawing area and form networks (→ Rules for SFC Connections). The procedure is identical with that for function block diagrams. The following special functions are also available:

To automatically create a step from a transition

- Point to an unconnected transition input or output.
- Drag the connection line into a free space within the Drawing Area.
- Release the mouse button. A step element is created at the position of the mouse pointer and connected to the transition.

To automatically create a transition from a step

- Point to an unconnected step input or output.
- Drag the connection line into a free space within the Drawing Area.
- Release the mouse button. A transition is created at the position of the mouse pointer and connected to the step element.

12.2 Rules for SFC Connections

When connecting the individual SFC elements to form an SFC network, observe the following rules:

- Each network begins with an initial step.
- A step output ends at a transition input.
- **At the beginning of an SFC branch:** A step output ends at several transition inputs.
- **At the end of a simultaneous branch:** Multiple step outputs end at a transition input.
- A transition output ends at a step input.
- **At the beginning of simultaneous branch:** A transition output ends at multiple step inputs.
- **At the end of an SFC branch:** Multiple transition outputs end at a step input.

12.3 Initial Step

The initial step is the first step of a step sequence and is active at the beginning of a program run. Each step sequence must start with an initial step and may only contain one initial step.

To create an initial step

- Open the Blocks tab located in the **Object Panel**.
- Click the column header **Library Type** to sort the table content by *SFC Element*.
- Drag the *Initial Step* element onto the Drawing Area. The initial step name is automatically created and is composed of S_ followed by a continuous number.
- Right-click the initial step just created and located in the Drawing Area and select **Properties** from the context menu. If required, enter a new name.
Alternatively: Double-click the element and enter the name directly.



Notice that the *Initial Step* parameter was automatically activated. The difference between an initial step and a normal SFC step is the wider and darker border of the initial step.

12.4 Step

SFC steps describe states within SFC networks and are represented by graphical symbols in the Drawing Area. The SFC input is located on the upper part, the SFC output on the lower part. SFC inputs and outputs together with transitions are used to create the connection to the SFC network. Two SFC steps may not be directly connected to one another!

The step flag output (= FBD output) to which an Transition, an Action or an FBD logic can be connected, is located on the right side of the symbol. The step flag output is set to TRUE if the SFC step is active. Normally, the step flag output is connected to an SFC action.

12.4.1 Definition According to IEC 61131-3

A step corresponds to a situation in which the behavior of one POU (with respect to its inputs and outputs) follows a number of rules that are defined by the actions associated with the step. A step is either active or inactive; at any given point in time, the POU state is defined by the quantity of active steps and the values of the internal and output variables.

To create an SFC step

- Open the Blocks tab located in the **Object Panel**.
- Click the column header **Library Type** to sort the table content by *SFC Element*.
- Drag the *Action* element onto the Drawing Area. The initial step name is automatically created and is composed of **S_** followed by a continuous number.
- Right-click the step just created and located in the Drawing Area and select **Properties** from the context menu. If required, enter a new name.
Alternatively: Double-click the graphical object and enter the name directly.



Keep in mind that the *Initial Step* parameter is deactivated. Activate an *Initial Step* to transform an SFC step into an Initial Step. The difference between an SFC step and an initial step is the darker border color of the SFC step.

To evaluate the SFC step state and the elapsed SFC step time

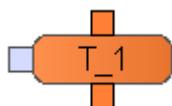
- Create an SFC step such as described [above](#).
- Open the Instances tab located in the **Object Panel**.
- Click the title of the **Data Type** column to sort the table contents by *Step*.
- In the *Name* column, search for the step just created (default value **S_n**) and click the plus sign next to the name. The instance variables **S_n.t** and **S_n.x** are displayed.
- Drag both instance variables onto the Drawing Area.
- **Step State:** Evaluate the SFC step state using the Boolean variable **S_n.x**. If the step is active, the variable is TRUE, otherwise it is FALSE.
- **Elapsed SFC Step Time:** Evaluate the elapsed SFC step time using the TIME variable **S_n.t**. If the SFC step is active or was once active, the variable value is greater than t#0ms. If the SFC step has not yet been activated, the value is equal to t#0ms.

12.5 Transition

SFC transitions describe state changes and are represented in the Drawing Area by graphical symbols. The SFC input is located on the upper part of the graphical symbol, the SFC output on the lower part.

The input for the transition condition is located on the left side. This input is of BOOL data type and can be connected to variables, constants and other POU outputs. As long as the transition condition is FALSE, the transition is not connected.

The transition name is placed within the symbol.



12.5.1 Definition According to IEC 61131-3

A transition specifies the condition under which the control passes from one or multiple steps preceding the transition to one or multiple steps following the transition along the corresponding connection. Each transition must have a corresponding transition condition.

To create an SFC transition

- Open the Blocks tab located in the **Object Panel (Engineering Kit)**.
- Click the column header **Library Type** to sort the table content by *SFC Element*.
- Drag the *Transition* element onto the Drawing Area. The transition name is automatically created and is composed of T_ followed by a continuous number.
- **Alternatively:** Drag a connection line from an SFC step onto any empty area within the Drawing Area. Release the mouse button to create an SFC transition at the position of the mouse pointer.
- Right-click the transition just created and located in the Drawing Area and select **Properties** from the context menu. If required, enter a new name.
- **Alternatively:** Double-click the element and enter the name directly.
- If required, enter a new value in the *Manual Execution Order* field. This action overrides the automatic execution order of SILworX. The lower the value, the higher the priority of the SFC transition. The default value is 0 and means the lowermost priority.

12.6 Action

SFC actions contain functions that should expire while an SFC step is active. The function behavior is defined by the action qualifier. This standardized acronym can be used, for instance, to define if a system part should be started immediately or with a certain delay.

Boolean variables are set or reset in SFC actions such that they can also be used in combination with any FBD logic. However, an SFC action input is usually connected directly to the output of an SFC step.

Graphical symbols are used to represent SFC actions in the Drawing Area. The input is located on the left side of the graphical symbol. The input is always of BOOL data type. The output is located on the right side of the graphical symbol.

12.6.1 SFC Action Fields



The graphical symbol of an SFC action contains the following fields:

- Action Qualifier (AQ)
- Time Period (T) (at the bottom left-hand side, it is only displayed for action qualifiers with delay time)
- SFC Action Name (AN) (on the right)

12.6.2 Behavior of SFC Actions

The SFC action output is TRUE if the function defined by the action qualifier is also TRUE. The following points must be observed:

- If an SFC action is used multiple times with an identical name, and the action qualifier R (overriding reset) is set during one usage, the output is set to FALSE for all usages when the overriding reset is set to TRUE. R has the effect to overwrite all the other action qualifiers.
For the diagrams see: [N+R](#), [S+R](#), [L+R](#), [D+R](#), [P+R](#), [SD+R](#), [DS+R](#), [SL+R](#), [P0+R](#), [P1+R](#), [PN+R](#).
- If an SFC action is used multiple times with an identical name, the outputs of all usages are disjuncted (OR gate). Qualifiers with a time delay are exceptions.
- If an SFC action is used multiple times with the identical name and a qualifier with a time delay (L, D, SD, DS, SL, PN), and several usages are active simultaneously, none of the usages are processed.

12.6.3 Functions and SFC Elements

If both step sequences and FBD are used in a logic, the sequence networks are processed first, and then the FBD networks (→ Show Network/Execution Order).

Notice that functions in accordance with IEC 61131-3 have no storing capability, i.e., the result can longer be used within the function after the function was called. An example is provided in the List of the Standard Function Blocks.

12.6.4 Differences Between ELOP II and SILworX

Compared to ELOP II, the resulting logic behaves differently in several instances and must be taken into account, particularly when routines written for ELOP II are used again:

- In SILworX, functions do not save results. Values from the previous cycle are thus not available and the SFC actions may behave differently than in ELOP II applications.
- Compared to ELOP II, the network and processing sequence can be different. Depending on the function result, if an action should be run, a variable must be added between function and action. In doing so, the required saving behavior is achieved.

To create an SFC action

- Open the Blocks tab located in the **Object Panel (engineering kit)**.
- Click the column header **Library Type** to sort the table content by *SFC Element*.
- Drag the *Action* element onto the Drawing Area.
The SFC Action Name (AN) is automatically created; it is composed of the prefix *Action_* and a consecutive number.
The default action qualifier is N.
- Right-click the SFC action just created and located in the Drawing Area and select **Properties** from the context menu. Set the parameters Action Qualifier (AQ), Time Period (T) and Name.
- **Alternatively:** Double-click the action qualifier in the SFC action and select another action qualifier from the drop down-list. The time for action qualifiers with time delay can also be directly modified from within the SFC action.

To use the output of a SFC action at another logic position:

- Click the **Instances** tab located in the Object Panel. If required, sort the instances by the *Action* data type.
- In the *Name* column, click the plus sign next to the name of the output variable. The Q value field of BOOL data type is displayed.
- Copy Q dragging it from the Object Panel to the required location.
Alternatively: Select **Create Value** field from the context menu of the FBD Editor and enter the name of the output variable in the value field, e. g., *A_1.Q*. Notice that the name is case sensitive.

12.6.5 SFC Action Name (AN)

The SFC action name is the unique identifier of an SFC action. The SFC action name represents a Boolean variable of type VAR_ACTION as displayed in the **Instance Variables** tab of the Object Panel (Engineering Kit).

The action name is automatically assigned when a new SFC action is created. The SFC action name is composed of the prefix *Action_* and a consecutive number. If required, the default value can be modified.

Use the variables *SFC Action Name.trig* and *SFC Action Name.state* to directly access the input and output value of an SFC action.

12.7 Creating SFC Connectors

An SFC connector represents the logical connection between a data source and a data sink. The SFC input connector is connected to the data source, the SFC output connector to the the data sink. Unlike FBD connectors, only one SFC output connector can be associated with an SFC input connector. Due to the 1:1 relationship between SFC input and output connectors, simultaneous nodes and branching nodes cannot be hidden.

To create an SFC input connector

- Right-click a free space within the Drawing Area and select **SFC Connector**. An SFC input connector with the default name **SFC_Connector_n** is created instead of the mouse pointer, where n represents a continuous number.
- **Renaming:** Double-click the SFC input connector. A text field for entering a new name appears.
- Connect the input connector with a connection line of the SFC logic.

To create an SFC output connector

- Open the **Connectors** tab located in the Object Panel (engineering kit).
- Drag the SFC output connector onto a free space within the Drawing Area.
- Connect the output connector with a connection line of the SFC logic.

12.8 Action Qualifier (AQ)

The action qualifier is selected to define when the instruction should be performed or which value should be assigned to the connected Boolean variable.

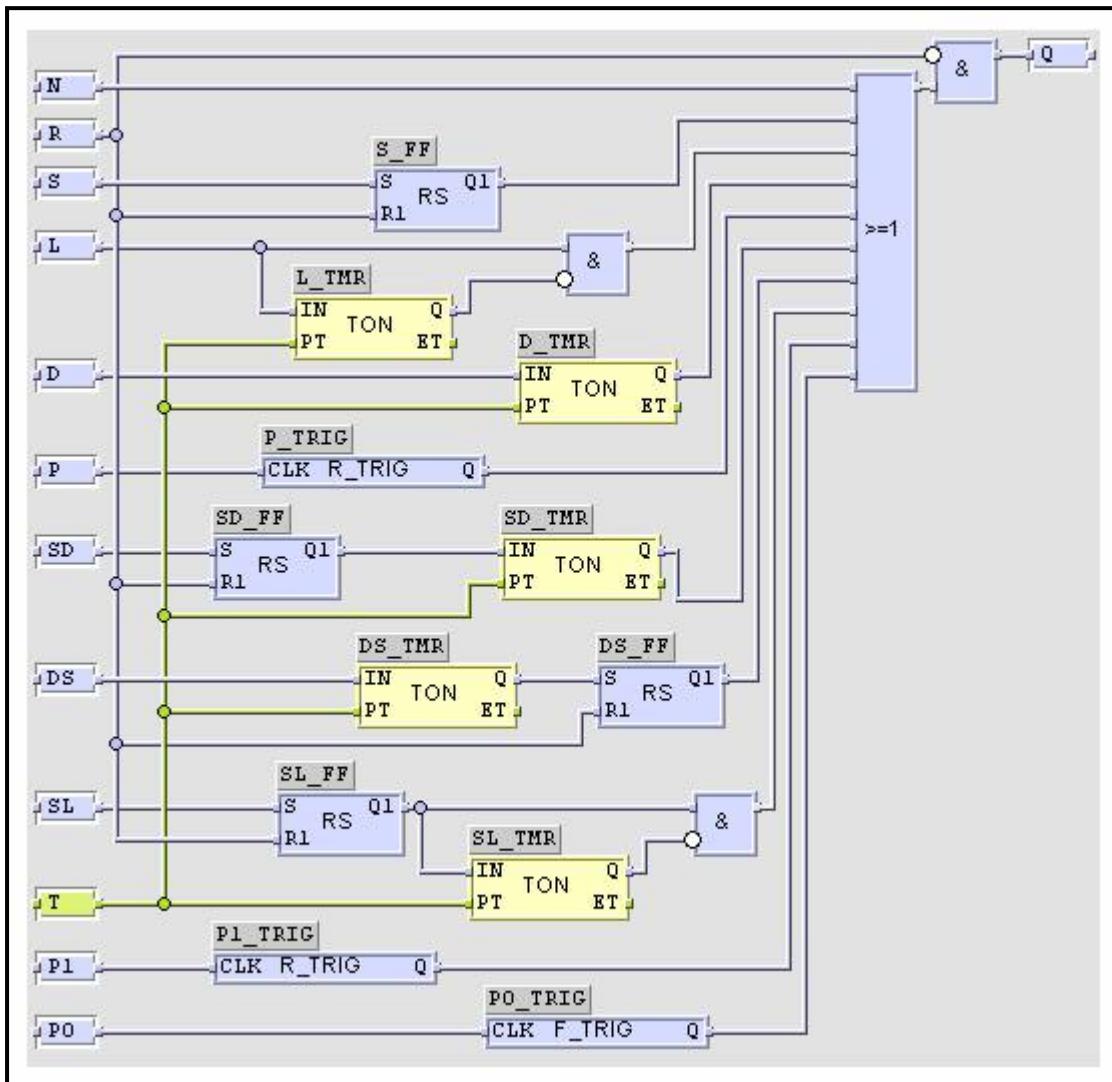
The action qualifier is either configured in the **Properties** of the SFC action or by double-clicking the field with the action qualifier within the graphical symbol. The following action qualifiers can be selected from the drop-down list:

AQ	Short name	Description	
(empty)		Same as N	
N	Non-stored	Input	Output
		TRUE	The output is TRUE as long as the input is TRUE (→ Action Qualifier N+R).
		FALSE	FALSE
R	Overriding Reset	Input	Output
		TRUE	The output is set to FALSE. This function has priority with respect to other action qualifiers.
		FALSE	No change
S	Set Stored	Input	Output
		TRUE	The output is set to TRUE. The state is saved (→ Action Qualifier S+R).
		FALSE	TRUE
L	Time Limited	Input	Output
		TRUE	The output is set to TRUE for the defined duration. Once the time period has expired, the output is set to FALSE (→ Action Qualifier L+R).
		FALSE	FALSE
D	Time Delayed	Input	Output
		TRUE	The output is only set to TRUE if the predefined delay has expired (→ Action Qualifier D+R).
		FALSE	FALSE
P	Pulse (Edge)	Input	Output
		TRUE	No change
		FALSE → TRUE	The output is set to TRUE if the edge at the input changes from FALSE to TRUE (→ Action Qualifier P+R).

AQ	Short name	Description	
		Input	Output
SD	Stored and Time-Delayed	TRUE	The output is only set to TRUE if the predefined delay has expired (→ Action Qualifier SD+R).
		TRUE → FALSE	The output is also set to TRUE if the input is set to FALSE during the delay.
DS	Delayed and Stored	TRUE	Once the predefined delay has expired, the output is only set to TRUE if the input was continuously set to TRUE during the entire delay. The output state is saved (→ Action Qualifier DS+R).
		TRUE → FALSE	The output remains set to TRUE if the input changes from FALSE to TRUE during the delay.
SL	Stored and Time-Limited	TRUE	The output is set to TRUE for the duration of the predefined delay (→ Action Qualifier SL+R).
		FALSE	No change
P0	Pulse (Falling Edge)	Detection of a falling edge. The function is identical with the function of the EDGE Sub-Library function block (→ Action Qualifier P0+R).	
P1	Pulse (Rising Edge)	Detection of a rising edge. The function is identical with the function of the R_TRIG function block (→ Action Qualifier P1+R).	
PN	Time Pulse, not Saved (HIMA-specific function)	TRUE	The output is set to TRUE as long as the input is TRUE and at least for the defined time period (→ Action Qualifier PN+R).
		FALSE	No change

12.8.1 Function of the Action Qualifier

The following graphic shows the function of the qualifier for an SFC action.



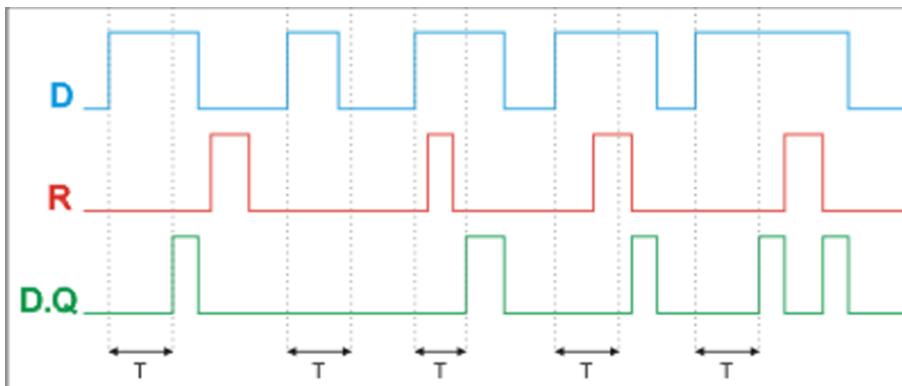
An SFC action with R qualifier triggers a reset and inhibits the effect of the SFC action.

12.8.2 Time Period (T)

If a qualifier requiring a time parameter was selected for an SFC action, the field for indicating the time is displayed in the SFC action symbol. A valid time value must be entered for the following action qualifiers:

- Time delayed: D, SD, DS
- Time limited: L, SL
- Time pulse: PN

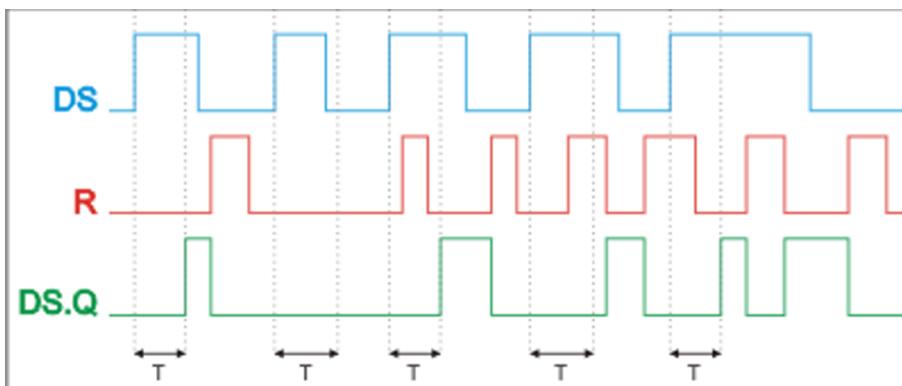
12.8.3 Action Qualifier D+R



If the input is continuously TRUE, the *D.Q* output of the SFC action (D) is set to TRUE upon expiration of the time period defined at *T*. The decisive element is the value of *T* at the moment of the positive edge at L. As soon as the input is FALSE, *D.Q* is also set to FALSE.

An SFC action (R) with identical name triggers a reset and does not affect the time period.

12.8.4 Action Qualifier DS+R

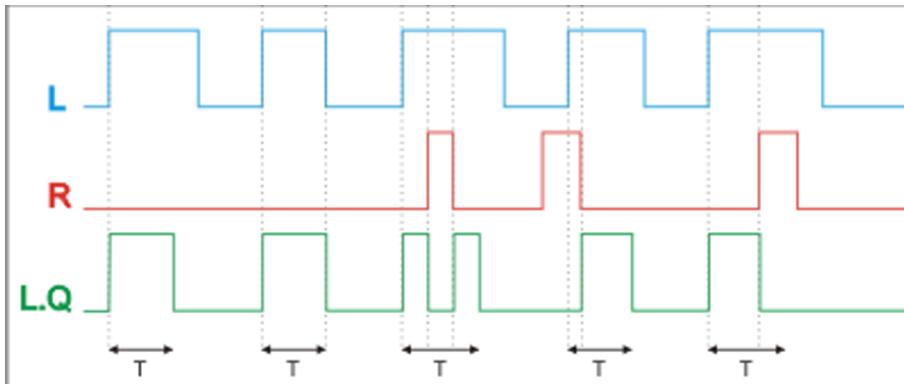


An SFC action (DS) corresponds to a serial connection of two SFC actions with the action qualifiers D and S. First the D function is processed, then the S function.

The value of *T* defined at the moment of the rising edge at DS.

An SFC action (R) with identical name triggers a reset and inhibits the effect of the SFC action (DS).

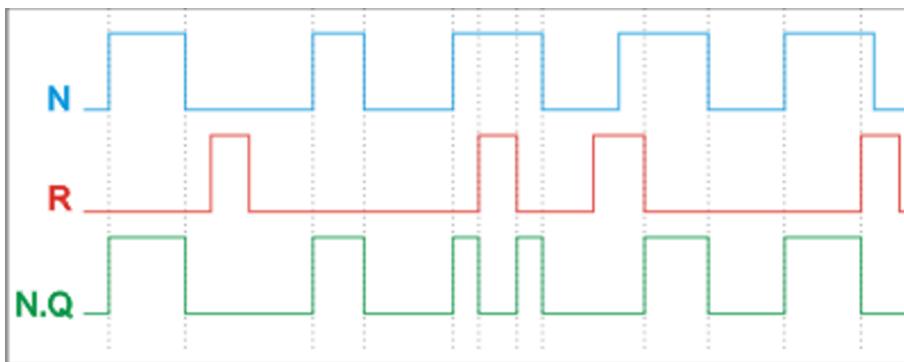
12.8.5 Action Qualifier L+R



If the input is set to TRUE, the *L.Q* output of the SFC action (L) is set to TRUE for the duration of the time defined with *T*. The decisive element is the value of *T* at the moment of the positive edge at L.

An SFC action (R) with identical name triggers a reset and does not affect the time period.

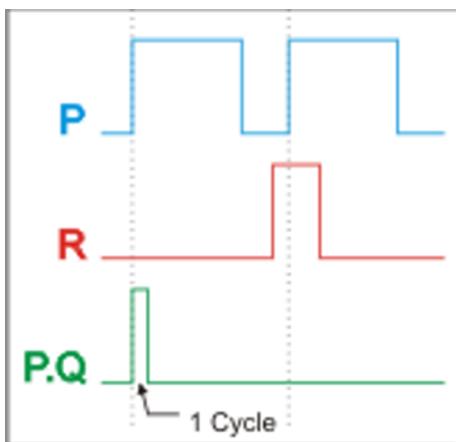
12.8.6 Action Qualifier N+R



The *N.Q* output of the SFC action (N) is set to TRUE if the input is set to TRUE.

If the SFC action (R) is used again with an identical action name, a TRUE signal triggers a reset of the output when the action is used the second time.

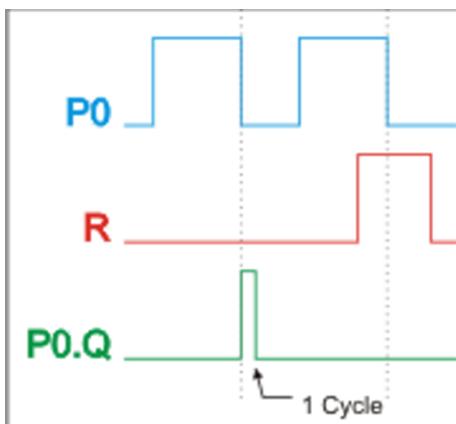
12.8.7 Action Qualifier P+R



After a rising edge at the input, the *P.Q* output of the SFC action (P) is set to TRUE for the duration of a user program cycle.

If the SFC action (R) is used again with an identical action name, a TRUE signal prevents the detection of the rising edge when the action is used the second time.

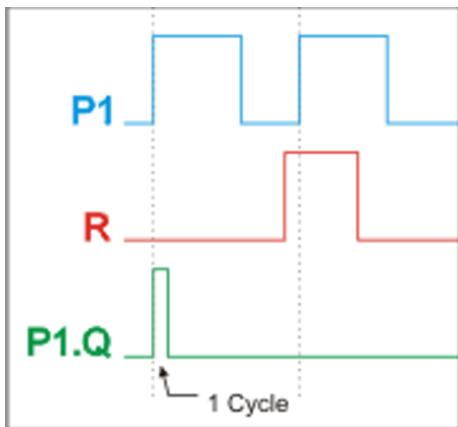
12.8.8 Action Qualifier P0+R



After a falling edge at the input, the *P0.Q* output of the SFC action (P0) is set to TRUE for the duration of a user program cycle.

If the SFC action (R) is used again with an identical action name, a TRUE signal prevents the detection of the falling edge when the action is used the second time.

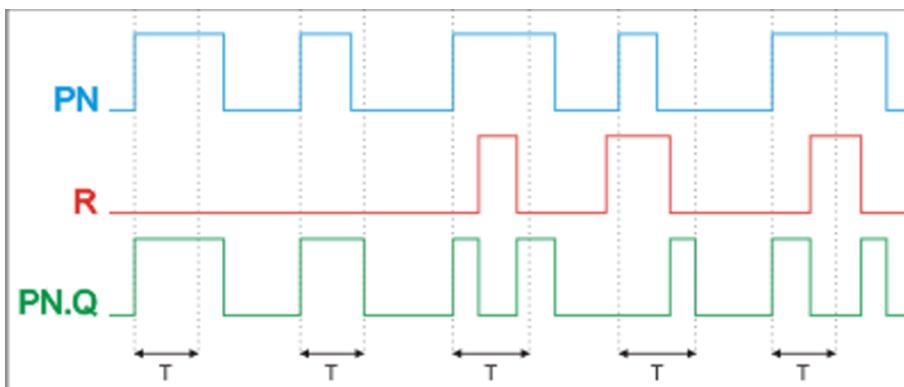
12.8.9 Action Qualifier P1+R



After a rising edge at the input, the **P1.Q** output of the SFC action (P) is set to TRUE for the duration of a user program cycle.

If the SFC action (R) is used again with an identical action name, a TRUE signal prevents the detection of the rising edge when the action is used the second time.

12.8.10 Action Qualifier PN+R

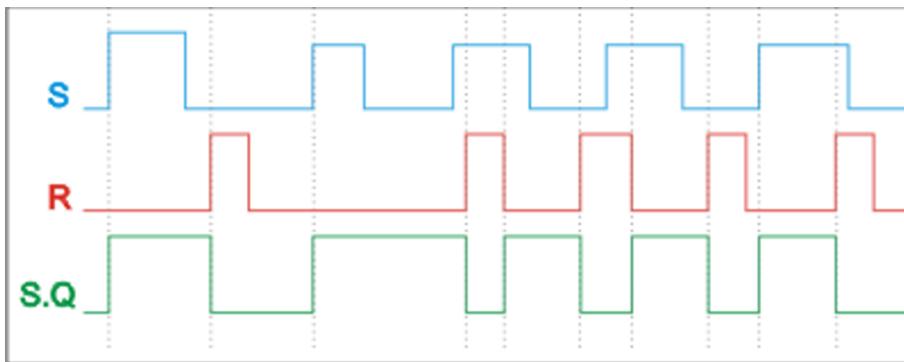


The **PN.Q** output of the SFC action (PN) is set to TRUE if the input is set to TRUE. The output remains set to TRUE for at least the duration of **T**, even if in the meantime the input is set to FALSE.

The value of **T** defined at the moment of the rising edge at PN.

An SFC action (R) with identical name triggers a reset and inhibits the effect of the SFC action (PN).

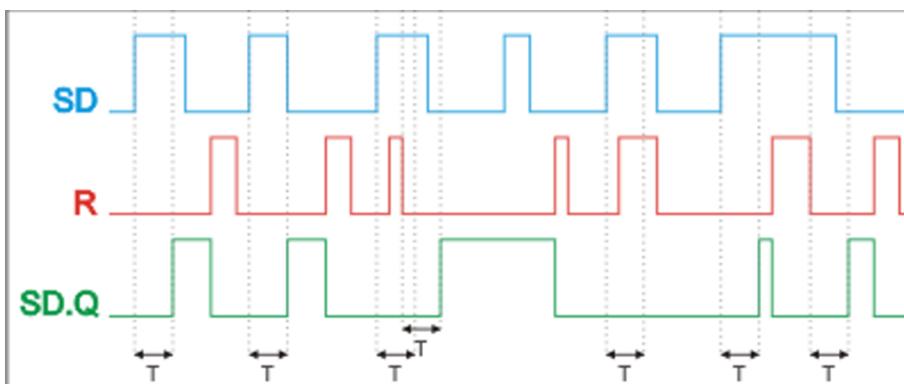
12.8.11 Action Qualifier S+R



The S.Q output of the SFC action (S) is set to TRUE if the input is set to TRUE.

To reset the output to FALSE, perform the SFC action once again with the identical action name and assign it the R action qualifier. As soon as the input is set to TRUE after the second usage, the S.Q output is set to FALSE.

12.8.12 Action Qualifier SD+R

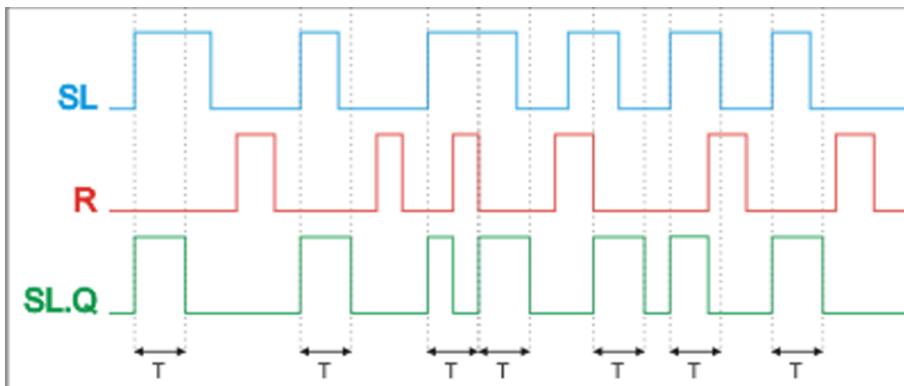


An SFC action (SD) corresponds to a serial connection of two SFC actions with the action qualifiers and S and D. First the S function is processed, then the D function.

The value used is that of T defined at the moment of the rising edge at SD or of the falling edge at R (t4, t7).

An SFC action (R) with identical name triggers a reset and inhibits the effect of the SFC action (SD).

12.8.13 Action Qualifier SL+R



An SFC action (SL) corresponds to a serial connection of two SFC actions with the action qualifiers *S* and *L*. First the *S* function is processed, then the *L* function.

The value used is that of *T* defined at the moment of the rising edge at SL or of the falling edge at R (t4, t5).

An SFC action (R) with identical name triggers a reset and inhibits the effect of the SFC action (SL).



A TRUE signal at the input of the SFC action (SL) is ignored if no reset has occurred since the last TRUE signal (see t5+T through t6).

HI 800 717 E

© 2015 HIMA Paul Hildebrandt GmbH

HIMA Paul Hildebrandt GmbH

Albert-Bassermann-Str. 28 | 68782 Brühl, Germany

Phone +49 06202 709-0 | Fax +49 06202 709-107

info@hima.com | www.hima.com



SAFETY
NONSTOP

