



PLC500N CPU180-X

LIST OF FIRMWARE FUNCTIONS AND FUNCTION BLOCKS

Bitwise Boolean functionPage 2

AND

OR
NOT
XOR

Arithmetic functionsPage 4

ADD

SUB
MUL
DIV
ABS
SQRT
EXP
EXPT
LN
LOG
MOD
MOVE
NEG

Function BlocksPage 8

TON

TOF
TP
CTD
CTU
CTUD
F_TRIG
R_TRIG
RS
SR



Selection functionsPage 13

LIMIT

- MAX
- MIN
- SEL

Trigonometric functions.....Page 15

ACOS

- ASIN
- ATAN
- COS
- SIN
- TAN

Comparison functions.....Page 17

EQ

- NE
- GT
- GE
- LT
- LE

Bit-String functions.....Page 20

SHL

- SHR
- ROL
- ROR

Type conversion functions.....Page 22



Bitwise Boolean function

AND

Description

This bitwise Boolean function realizes a logical **AND** integration of the operands connected to the input parameters.

<u>Parameter</u>	<u>Data types</u>	<u>Description</u>
IN1	ANY_BIT	input value
IN2	ANY_BIT	input value
OUT	ANY_BIT	output value

Note: The input IN2 can be duplicated. All parameters can be negated.

Note: All parameters must have the same data type.

OR

Description

This bitwise Boolean function realizes a logical OR-integration of the operands connected to the input parameters.

<u>Parameter</u>	<u>Data types</u>	<u>Description</u>
IN1	ANY_BIT	first input value
IN2	ANY_BIT	second input value
OUT	ANY_BIT	output value

Note: The input IN2 can be duplicated. All parameters can be negated.

Note: All parameters must have the same data type.

NOT

Description

This bitwise Boolean function negates bitwise the operand connected to the input parameter.

<u>Parameter</u>	<u>Data types</u>	<u>Description</u>
IN1	ANY_BIT	input value
OUT	ANY_BIT	output value

Note: All parameters must have the same data type.



XOR

Description

This bitwise Boolean function realizes a logical exclusive-OR-integration of the operands connected to the input parameters.

<u>Parameter</u>	<u>Data types</u>	<u>Description</u>
IN1	ANY_BIT	first input value
IN2	ANY_BIT	second input value
OUT	ANY_BIT	output value

Note: The input IN2 can be duplicated. All parameters can be negated.

Note: All parameters must have the same data type.



Arithmetic functions

ADD

Description

This arithmetic function realizes an **addition** of the operands connected to the input parameters.

<u>Parameter</u>	<u>Data types</u>	<u>Description</u>
IN1	ANY_NUM	summand
IN2	ANY_NUM	summand
OUT	ANY_NUM	sum

Note: The input IN2 can be duplicated.

Note: All parameters must have the same data type.

SUB

Description

This arithmetic function **subtracts** IN2 from IN1.

<u>Parameter</u>	<u>Data types</u>	<u>Description</u>
IN1	ANY_NUM	minuend
IN2	ANY_NUM	subtrahend
OUT	ANY_NUM	difference

Note: All parameters must have the same data type.

MUL

Description

This arithmetic function multiplies the operands connected to the input parameters.

<u>Parameter</u>	<u>Data types</u>	<u>Description</u>
IN1	ANY_NUM	first input value
IN2	ANY_NUM	second input value
OUT	ANY_NUM	output value

Note: The input IN2 can be duplicated.

Note: All parameters must have the same data type.



DIV

Description

This arithmetic function **divides** the operand connected to IN1 by the operand connected to IN2.

<u>Parameter</u>	<u>Data types</u>	<u>Description</u>
IN1	ANY_NUM	dividend
IN2	ANY_NUM	divisor
OUT	ANY_NUM	output value

Note: All parameters must have the same data type.

ABS

Description

This arithmetic function calculates the **absolute** value of the operand connected to the input parameter.

<u>Parameter</u>	<u>Data types</u>	<u>Description</u>
IN1	ANY_NUM	input value
OUT	ANY_NUM	output value

Note: All parameters must have the same data type.

SQRT

Description

This arithmetic function calculates the **square root** of the operand connected to the input parameter.

<u>Parameter</u>	<u>Data types</u>	<u>Description</u>
IN1	REAL	input value
OUT	REAL	output value



EXP

Description

This arithmetic function calculates the **natural exponential** of the operand connected to the input parameter.

<u>Parameter</u>	<u>Data types</u>	<u>Description</u>
IN1	REAL	exponent to base e
OUT	REAL	output value

EXPT

Description

This arithmetic function realizes an exponentiation.

<u>Parameter</u>	<u>Data types</u>	<u>Description</u>
IN1	ANY_REAL	base
IN2	ANY_INT	exponent
OUT	ANY_REAL	output

LN

Description

This arithmetic function calculates the **natural logarithm** of the operand connected to the input parameter.

<u>Parameter</u>	<u>Data types</u>	<u>Description</u>
IN1	REAL	input value
OUT	REAL	output value

LOG

Description

This arithmetic function calculates the **logarithm to the base of 10** of the operand connected to the input parameter.

<u>Parameter</u>	<u>Data types</u>	<u>Description</u>
IN1	REAL	input value
OUT	REAL	output value



MOD

Description

This arithmetic function divides the operand connected to the input IN1 by the operand connected to input IN2 and returns the **remainder** of the division at OUT.

<u>Parameter</u>	<u>Data types</u>	<u>Description</u>
IN1	ANY_INT	dividend
IN2	ANY_INT	divisor
OUT	ANY_INT	remainder

Note: All parameters must have the same data type.

MOVE

Description

This arithmetic function **moves** the value of the operand connected to the input parameter to the operand connected to the output parameter.

<u>Parameter</u>	<u>Data types</u>	<u>Description</u>
IN	ANY_NUM	input value
OUT	ANY_NUM	output value

Note: All parameters must have the same data type.

NEG

Description

This arithmetic function **negates** the operand connected to the input parameter.

<u>Parameter</u>	<u>Data types</u>	<u>Description</u>
IN1	ANY_NUM	input value
OUT	ANY_NUM	output value

Note: All parameters must have the same data type.



Function Blocks

TOF

Description

This timer function block realizes an **off-delay timing**.

If the input IN changes from TRUE to FALSE, switching off is delayed for the time interval at input PT. After PT has passed, FALSE is issued at the output Q. The time which has already elapsed is indicated at the output ET.

<u>Parameter</u>	<u>Data types</u>	<u>Description</u>
IN	BOOL	If a falling edge is detected, the off-delay timing is started.
PT	TIME	preset time interval for the delay
Q	BOOL	TRUE if IN = TRUE and ET < PT FALSE if IN = FALSE and ET >= PT
ET	TIME	elapsed time interval

Note: The input IN and the output Q can be negated.

TON

Description

This timer function block realizes an **on-delay timing**.

If the input IN changes from FALSE to TRUE, switching on is delayed for the time interval at input PT. After PT has passed, TRUE is issued at the output Q. The time which has already elapsed is indicated at the output ET.

<u>Parameter</u>	<u>Data types</u>	<u>Description</u>
IN	BOOL	If a rising edge is detected, the on-delay timing is started.
PT	TIME	preset time interval for the delay
Q	BOOL	TRUE if IN = TRUE and ET < PT FALSE if IN = FALSE and ET >= PT
ET	TIME	elapsed time interval

Note: The input IN and the output Q can be negated.



TP

Description

This timer function block creates a **pulse**.

If the input IN changes from FALSE to TRUE, a pulse is created at output Q for the time interval PT. The time which has already elapsed is indicated at the output ET. If IN gets TRUE for another time and PT has not been elapsed, it has no impact on the duration of the pulse created at output Q.

<u>Parameter</u>	<u>Data types</u>	<u>Description</u>
IN	BOOL	If a rising edge is detected, a pulse is created.
PT	TIME	preset time interval for the pulse
Q	BOOL	TRUE if IN = TRUE and ET < PT FALSE if IN = FALSE and ET >= PT
ET	TIME	elapsed time interval

Note: The input IN and the output Q can be negated.

CTD

Description

This counter function block **counts down**. In case of a rising edge at the input CD and LOAD = FALSE, CV decrements by one. If the final value of the counter (0) is achieved, TRUE is issued at the output Q and the function block stops counting.

If LOAD = TRUE, the counter is initialized by the value of the input PV. To enable the counting process, the input LOAD must be FALSE. Otherwise the counter will always be re-initialized.

<u>Parameter</u>	<u>Data types</u>	<u>Description</u>
CD	BOOL	If a rising edge is detected, CV decrements by one.
LOAD	BOOL	If TRUE, the counter is initialized with PV. If FALSE, counting is enabled.
PV	INT	preset value
Q	BOOL	TRUE if CV = 0
CV	INT	counter result

Note: The output Q can be negated.



CTU

Description

This counter function block counts up. In case of a rising edge at the input CU and RESET = FALSE, CV is increased by one. If the final value of the counter (PV) is achieved, TRUE is issued at the output Q and the function block stops counting.

If RESET = TRUE, the counter is initialized with 0. To enable the counting process, the input RESET must be FALSE. Otherwise the counter will always be re-initialized.

Parameter	Data types	Description
CU	BOOL	If a rising edge is detected, CV is increased by one
RESET	BOOL	If TRUE, the counter is initialized with 0 If FALSE, counting is enabled
PV	INT	preset value
Q	BOOL	TRUE if CV = PV
CV	INT	counter result

Note: The output Q can be negated.

CTUD

Description

This counter function block counts up or down. In case of a rising edge at the input CU, CV is increased by one. In case of a rising edge at the input CD, CV decrements by one. If CV = PV, TRUE is issued at the output QU. If CV = 0, TRUE is issued at the output QD.

If the input RESET = TRUE, the counter is initialized with 0. If the input LOAD = TRUE, the counter is initialized with PV. To enable the counting process, the inputs RESET and LOAD must be FALSE. Otherwise the counter will always be re-initialized.

Parameter	Data types	Description
CU	BOOL	If a rising edge is detected, CV is increased by one.
CD	BOOL	If a rising edge is detected, CV decrements by one.
RESET	BOOL	If TRUE the counter is initialized with 0. If FALSE counting is enabled.
LOAD	BOOL	If TRUE the counter is initialized with PV. If FALSE counting is enabled.
PV	INT	preset value
QU	BOOL	TRUE if CV = PV
QD	BOOL	TRUE if CV = 0
CV	INT	counter result

Note: The outputs QU and QD can be negated.



F_TRIG

Description

This edge detection function block detects a **falling edge**. If a falling edge is detected at the input CLK, the output Q changes from FALSE to TRUE. Q remains TRUE until the next execution of the function block.

If the function block is called for the first time, Q is FALSE until the first edge is detected.

<u>Parameter</u>	<u>Data types</u>	<u>Description</u>
CLK	BOOL	detects a falling edge
Q	BOOL	If a falling edge is detected, Q changes from FALSE to TRUE

R_TRIG

Description

This edge detection function block detects a **rising edge**. If a rising edge is detected at the input CLK, the output Q changes from FALSE to TRUE. Q remains TRUE until the execution of the function block.

If the function block is called for the first time, Q is FALSE until the first edge is detected.

<u>Parameter</u>	<u>Data types</u>	<u>Description</u>
CLK	BOOL	detects a rising edge
Q	BOOL	If a rising edge is detected, Q changes from FALSE to TRUE

RS

Description

This bistable function block realizes a prior reset of the output Q1. If the input SET = TRUE, the output Q1 is set. Q1 remains set even if SET becomes FALSE. Q1 is reset, if RESET1 = TRUE. If both inputs = TRUE, the output Q1 is set by RESET1 to FALSE.

If the function block is called for the first time, Q1 is FALSE.

<u>Parameter</u>	<u>Data types</u>	<u>Description</u>
SET	BOOL	IF TRUE Q1 is set
RESET1	BOOL	IF TRUE Q1 is reset dominant
Q1	BOOL	output value

Note: All parameters can be negated.



SR

Description

This bistable function block realizes a prior set of the output Q1. If the input SET1 = TRUE, the output Q1 is set. Q1 remains set even if SET becomes FALSE. Q1 is reset, if RESET = TRUE. If both inputs = TRUE, the output Q1 is set by SET1 to TRUE.

If the function block is called for the first time, Q1 is FALSE.

<u>Parameter</u>	<u>Data types</u>	<u>Description</u>
SET1	BOOL	IF TRUE Q1 is set dominant
RESET	BOOL	If TRUE Q1 is reset
Q1	BOOL	output

Note: All parameters can be negated.



Selection functions

LIMIT

Description

This selection function **limits** the value of the input parameter IN to the value sector defined by the input parameters MN (minimum) and MX (maximum).

If $MN \leq IN \leq MX$ then $OUT = IN$.

If $IN < MN$ then $OUT = MN$.

If $IN > MX$ then $OUT = MX$.

<u>Parameter</u>	<u>Data types</u>	<u>Description</u>
MN	ANY_INT	minimum limit
IN	ANY_INT	input value
MX	ANY_INT	maximum limit
OUT	ELEMENTARY	output value

Note: All parameters must have the same data type.

MAX

Description

This selection function determines the **maximal** value of the operands connected to the input parameters.

Note: This overloaded function should be normally used. Only in case of the data type STRING or if memory problems appeared the typed function should be used.

<u>Parameter</u>	<u>Data types</u>	<u>Description</u>
IN1	ANY_INT	first input value
IN2	ANY_INT	second input value
OUT	ELEMENTARY	output value

Note: All parameters must have the same data type.



MIN

Description

This selection function determines the **minimal** value of the operands connected to the input parameters.

<u>Parameter</u>	<u>Data types</u>	<u>Description</u>
IN1	ANY_INT	first input value
IN2	ANY_INT	second input value
OUT	ELEMENTARY	output value

Note: All parameters must have the same data type.

SEL

Description

This selection function **selects** one of two inputs depending on the value of the first input G.

If G = FALSE, the value of the input IN0 is allocated to the output OUT.

If G = TRUE, the value of the input IN1 is allocated to the output OUT.

Note: This overloaded function should be normally used. Only in case of the data type STRING or if memory problems appeared the typed function should be used.

<u>Parameter</u>	<u>Data types</u>	<u>Description</u>
G	BOOL	selecting input
IN0	ANY_INT	input value
IN1	ANY_INT	input value
OUT	ANY	output value

Note: The input G can be negated.

Note: IN0, IN1 and OUT must have the same data type.



Trigonometric functions

ACOS

Description

This trigonometric function calculates the principal **arc cosine** of the operand connected to the input parameter.

<u>Parameter</u>	<u>Data types</u>	<u>Description</u>
IN1	REAL	input value
OUT	REAL	output value in radians

ASIN

Description

This trigonometric function calculates the principal **arc sine** value of the operand connected to the input parameter.

<u>Parameter</u>	<u>Data types</u>	<u>Description</u>
IN1	REAL	input value
OUT	REAL	output value in radians

ATAN

Description

This trigonometric function calculates the principal **arc tangent** value of the operand connected to the input parameter.

<u>Parameter</u>	<u>Data types</u>	<u>Description</u>
IN1	REAL	input value
OUT	REAL	output value in radians

COS

Description

This trigonometric function calculates the **cosine** of the operands connected to the input parameter.

<u>Parameter</u>	<u>Data types</u>	<u>Description</u>
IN1	REAL	input value in radians
OUT	REAL	output value



SIN

Description

This trigonometric function calculates the **sine** of the operand connected to the input parameter.

<u>Parameter</u>	<u>Data types</u>	<u>Description</u>
IN1	EAL	input value in radians
OUT	EAL	output value

TAN

Description

This trigonometric function calculates the **tangent** of the operand connected to the input parameter.

<u>Parameter</u>	<u>Data types</u>	<u>Description</u>
IN1	REAL	input value in radians
OUT	REAL	output value



Comparison functions

EQ (EQUAL)

Description

This comparison function compares the operands connected to the input parameters in regard to **equality**. Comparison proceeds from left to right.

Parameter	Data types	Description
IN1	ELEMENTARY	first input value
IN2	ELEMENTARY	second input value
OUT	BOOL	TRUE if the inputs are equal. FALSE if the inputs are not equal.

Note: The inputs IN1 and IN2 can be duplicated.

Note: The output OUT can be negated.

NE (NOT EQUAL)

Description

This comparison function compares the operands connected to the input parameters in regard to **inequality**.

Parameter	Data types	Description
IN1	ANY_NUM	first input value
IN2	ANY_NUM	second input value
OUT	BOOL	TRUE if IN1 and IN2 are not equal. FALSE if IN1 and IN2 are equal.

Note: The output OUT can be negated.

Note: IN1 and IN2 must have the same data type.



GT (GREATER THAN)

Description

This comparison function compares the operands connected to the input parameters in regard to **greater**. Comparison proceeds from left to right.

Note: The comparison is always a signed comparison. Therefore the values for DWORD are limited to 31 bit, for WORD to 15 bit and for BYTE to 7 bit.

Note: For string comparison use the corresponding typed function.

<u>Parameter</u>	<u>Data types</u>	<u>Description</u>
IN1	ELEMENTARY	first input value
IN2	ELEMENTARY	second input value
OUT	BOOL	TRUE if IN1 is greater than IN2. FALSE if IN1 is equal or less than IN2.

Note: The inputs IN1 and IN2 can be duplicated. The output OUT can be negated.

Note: All inputs must have the same data type.

GE (GREATER / EQUAL)

Description

This comparison function compares the operands connected to the input parameters in regard to **greater/equal**. Comparison proceeds from left to right.

Note: The comparison is always a signed comparison. Therefore the values for DWORD are limited to 31 bit, for WORD to 15 bit and for BYTE to 7 bit.

Note: For string comparison use the corresponding typed function.

<u>Parameter</u>	<u>Data types</u>	<u>Description</u>
IN1	ELEMENTARY	first input value
IN2	ELEMENTARY	second input value
OUT	BOOL	TRUE if IN1 is greater or equal than IN2. FALSE if IN1 is less than IN2.

Note: The inputs IN1 and IN2 can be duplicated. The output OUT can be negated.

Note: All inputs must have the same data type.



LT (LESS THAN)

Description

This comparison function compares the operands connected to the input parameters in regard to **less**. Comparison proceeds from left to right.

Note: The comparison is always a signed comparison. Therefore the values for DWORD are limited to 31 bit, for WORD to 15 bit and for BYTE to 7 bit.

Note: For string comparison use the corresponding typed function.

<u>Parameter</u>	<u>Data types</u>	<u>Description</u>
IN1	ELEMENTARY	first input value
IN2	ELEMENTARY	second input value
OUT	BOOL	TRUE if IN1 is lower than IN2. FALSE if IN1 is greater or equal than IN2.

Note: The output OUT can be negated.

LE (LESS / EQUAL)

Description

This comparison function compares the operands connected to the input parameters in regard to **less/equal**. Comparison proceeds from left to right.

Note: The comparison is always a signed comparison. Therefore the values for DWORD are limited to 31 bit, for WORD to 15 bit and for BYTE to 7 bit.

Note: For string comparison use the corresponding typed function.

<u>Parameter</u>	<u>Data types</u>	<u>Description</u>
IN1	ELEMENTARY	first input value
IN2	ELEMENTARY	second input value
OUT	BOOL	TRUE if IN1 is less or equal than IN2. FALSE if IN1 is greater than IN2.

Note: The output OUT can be negated.



Bit-String Functions

SHL

Description

This bit-string function realizes a **bitwise left shifting** of the operand connected to the input IN. N specifies the number of bits to be shifted. The empty bit positions are filled with zeros.

<u>Parameter</u>	<u>Data types</u>	<u>Description</u>
IN	ANY_BIT	input value
N	ANY_INT	number of bits to be shifted
OUT	ANY_BIT	output value

SHR

Description

This bit-string function realizes a **bitwise right shifting** of the operand connected to the input IN. N specifies the number of bits to be shifted. The empty bit positions are filled with zeros.

A bitwise right shifting of 3 bits is shown in the following figure:

<u>Parameter</u>	<u>Data types</u>	<u>Description</u>
IN	NY_BIT	input value
N	NY_INT	number of bits to be shifted
OUT	NY_BIT	output value

ROL

Description

This bit-string function realizes a **bitwise left rotation** of the operand connected to the input IN. N specifies the number of bits to be rotated.

A bitwise left rotation of 3 bits is shown in the following figure:

<u>Parameter</u>	<u>Data types</u>	<u>Description</u>
IN	ANY_BIT	input value
N	ANY_INT	number of bits to be rotated
OUT	ANY_BIT	output value



ROR

Description

This bit-string function realizes a **bitwise right rotation** of the operand connected to the input IN. N specifies the number of bits to be rotated.

A bitwise right rotation of 3 bits is shown in the following figure:

<u>Parameter</u>	<u>Data types</u>	<u>Description</u>
IN	ANY_BIT	input value
N	ANY_INT	number of bits to be rotated
OUT	ANY_BIT	output value



Type conversion functions

BOOL_TO_BYTE

Description

This type conversion function changes an input value of the data type BOOL into an output value of the data type BYTE.

If the input value is FALSE, the output value is converted to 16#00. The input value TRUE results in the output value 16#01.

<u>Parameter</u>	<u>Data type</u>	<u>Description</u>
IN	BOOL	input value
OUT	BYTE	output value

BOOL_TO_DINT

Description

This type conversion function changes an input value of the data type BOOL into an output value of the data type DINT.

If the input value is FALSE, the output value is converted to 0. The input value TRUE results in the output value 1.

<u>Parameter</u>	<u>Data type</u>	<u>Description</u>
IN	BOOL	input value
OUT	DINT	output value

BOOL_TO_DWORD

Description

This type conversion function changes an input value of the data type BOOL into an output value of the data type DWORD.

The input value is converted to the lowest-order bits of the output. The highest-order bits are set to zero. If the input value is FALSE, the output value is converted to 16#00000000. The input value TRUE results in the output value 16#00000001.

<u>Parameter</u>	<u>Data type</u>	<u>Description</u>
IN	BOOL	input value
OUT	DWORD	output value



BOOL_TO_INT

Description

This type conversion function changes an input value of the data type BOOL into an output value of the data type INT.

If the input value is FALSE, the output value is converted to 0. The input value TRUE results in the output value 1.

<u>Parameter</u>	<u>Data type</u>	<u>Description</u>
IN	BOOL	input value
OUT	INT	output value

BOOL_TO_REAL

Description

This type conversion function changes an input value of the data type BOOL into an output value of the data type REAL.

If the input value is FALSE, the output value is converted to 1.0000000 E+00. The input value TRUE results in the output value 0.0000000 E+00.

<u>Parameter</u>	<u>Data type</u>	<u>Description</u>
IN	BOOL	input value
OUT	REAL	output value

BOOL_TO_SINT

Description

This type conversion function changes an input value of the data type BOOL into an output value of the data type SINT.

If the input value is FALSE, the output value is converted to 0. The input value TRUE results in the output value 1.

<u>Parameter</u>	<u>Data type</u>	<u>Description</u>
IN	BOOL	input value
OUT	SINT	output value



BOOL_TO_UDINT

Description

This type conversion function changes an input value of the data type BOOL into an output value of the data type UDINT.

If the input value is FALSE, the output value is converted to 0. The input value TRUE results in the output value 1.

<u>Parameter</u>	<u>Data type</u>	<u>Description</u>
IN	BOOL	input value
OUT	UDINT	output value

BOOL_TO_UINT

Description

This type conversion function changes an input value of the data type BOOL into an output value of the data type UINT.

If the input value is FALSE, the output value is converted to 0. The input value TRUE results in the output value 1.

<u>Parameter</u>	<u>Data type</u>	<u>Description</u>
IN	BOOL	input value
OUT	UINT	output value

BOOL_TO_USINT

Description

This type conversion function changes an input value of the data type BOOL into an output value of the data type USINT.

If the input value is FALSE, the output value is converted to 0. The input value TRUE results in the output value 1.

<u>Parameter</u>	<u>Data type</u>	<u>Description</u>
IN	BOOL	input value
OUT	USINT	output value



BOOL_TO_WORD

Description

This type conversion function changes an input value of the data type BOOL into an output value of the data type WORD.

The input value is converted to the lowest-order bits of the output. The highest-order bits are set to zero. If the input value is FALSE, the output value is converted to 16#0000. The input value TRUE results in the output value 16#0001.

<u>Parameter</u>	<u>Data type</u>	<u>Description</u>
IN	BOOL	input value
OUT	WORD	output value

BYTE_TO_BOOL

Description

This type conversion function changes an input value of the data type BYTE into an output value of the data type BOOL.

Any input value unequal to 16#00 results in the output value TRUE.

<u>Parameter</u>	<u>Data type</u>	<u>Description</u>
IN	BYTE	input value
OUT	BOOL	output value

BYTE_TO_DINT

Description

This type conversion function changes an input value of the data type BYTE into an output value of the data type DINT.

A signed conversion is performed. For output data types greater than the input data type a sign-conformed expansion is additionally performed. For example the input value 16#80 results in the output value -128.

<u>Parameter</u>	<u>Data type</u>	<u>Description</u>
IN	BYTE	input value
OUT	DINT	output value



BYTE_TO_DWORD

Description

This type conversion function changes an input value of the data type BYTE into an output value of the data type DWORD.

The bit pattern of the input is converted to the lowest-order bits of the output. For example the input value 16#FF results in the output value 16#000000FF.

<u>Parameter</u>	<u>Data type</u>	<u>Description</u>
IN	BYTE	input value
OUT	DWORD	output value

BYTE_TO_INT

Description

This type conversion function changes an input value of the data type BYTE into an output value of the data type INT.

A signed conversion is performed. For output data types greater than the input data type a sign-conformed expansion is additionally performed. For example the input value 16#80 results in the output value -128.

<u>Parameter</u>	<u>Data type</u>	<u>Description</u>
IN	BYTE	input value
OUT	INT	output value

BYTE_TO_REAL

Description

This type conversion function changes an input value of the data type BYTE into an output value of the data type REAL.

The MSB (most significant bit) of the BYTE value is interpreted as sign bit. BYTE values greater than 16#7F result in negative values of the output data type. For example the input value 16#80 results in the output value -1.2800000 E+02.

<u>Parameter</u>	<u>Data type</u>	<u>Description</u>
IN	BYTE	input value
OUT	REAL	output value



BYTE_TO_SINT

Description

This type conversion function changes an input value of the data type BYTE into an output value of the data type SINT.

A signed conversion is performed. For output data types greater than the input data type a sign-conformed expansion is additionally performed. For example the input value 16#80 results in the output value -128.

<u>Parameter</u>	<u>Data type</u>	<u>Description</u>
IN	BYTE	input value
OUT	SINT	output value

BYTE_TO_UDINT

Description

This type conversion function changes an input value of the data type BYTE into an output value of the data type UDINT.

For example the input value 16#80 results in the output value 128.

<u>Parameter</u>	<u>Data type</u>	<u>Description</u>
IN	BYTE	input value
OUT	UDINT	output value

BYTE_TO_UINT

Description

This type conversion function changes an input value of the data type BYTE into an output value of the data type UINT.

For example the input value 16#80 results in the output value 128.

<u>Parameter</u>	<u>Data type</u>	<u>Description</u>
IN	BYTE	input value
OUT	UINT	output value



BYTE_TO_USINT

Description

This type conversion function changes an input value of the data type BYTE into an output value of the data type USINT.

For example the input value 16#80 results in the output value 128.

<u>Parameter</u>	<u>Data type</u>	<u>Description</u>
IN	BYTE	input value
OUT	USINT	output value

BYTE_TO_WORD

Description

This type conversion function changes an input value of the data type BYTE into an output value of the data type WORD.

The bit pattern of the input is converted to the lowest-order bits of the output. For example the input value 16#FF results in the output value 16#00FF.

<u>Parameter</u>	<u>Data type</u>	<u>Description</u>
IN	BYTE	input value
OUT	WORD	output value

DINT_TO_BOOL

Description

This type conversion function changes an input value of the data type DINT into an output value of the data type BOOL.

Any input value unequal to 0 results in the output value TRUE.

<u>Parameter</u>	<u>Data type</u>	<u>Description</u>
IN	DINT	input value
OUT	BOOL	output value



DINT_TO_BYTE

Description

This type conversion function changes an input value of the data type DINT into an output value of the data type BYTE.

The lowest-order bits of the input value are only converted to the output value. For example the input value -1 results in the output value 16#FF.

<u>Parameter</u>	<u>Data type</u>	<u>Description</u>
IN	DINT	input value
OUT	BYTE	output value

DINT_TO_DWORD

Description

This type conversion function changes an input value of the data type DINT into an output value of the data type DWORD.

<u>Parameter</u>	<u>Data type</u>	<u>Description</u>
IN	DINT	input value
OUT	DWORD	output value

DINT_TO_INT

Description

This type conversion function changes an input value of the data type DINT into an output value of the data type INT.

The lowest-order bits of the input value are only converted to the output value. For example the input value -1 results in the output value -1.

<u>Parameter</u>	<u>Data type</u>	<u>Description</u>
IN	DINT	input value
OUT	INT	output value



DINT_TO_REAL

Description

This type conversion function changes an input value of the data type DINT into an output value of the data type REAL.

<u>Parameter</u>	<u>Data type</u>	<u>Description</u>
IN	DINT	input value
OUT	REAL	output value

DINT_TO_SINT

Description

This type conversion function changes an input value of the data type DINT into an output value of the data type BYTE.

The lowest-order bits of the input value are only converted to the output value. For example the input value -2147483647 results in the output value -1.

<u>Parameter</u>	<u>Data type</u>	<u>Description</u>
IN	DINT	input value
OUT	SINT	output value

DINT_TO_UDINT

Description

This type conversion function changes an input value of the data type DINT into an output value of the data type UDINT.

The output values are always positive.

For example the maximum negative input value -2.147.483.648 results in the output value 2.147.483.648 and the input value -1 results in the output value 4.294.967.295 (maximum value of UDINT).

<u>Parameter</u>	<u>Data type</u>	<u>Description</u>
IN	DINT	input value
OUT	UDINT	output value



DINT_TO_UINT

Description

This type conversion function changes an input value of the data type DINT into an output value of the data type UINT.

The output values are always positive.

For example the input value -32768 results in the output value 32768 and the input value -1 results in the output value 65535 (maximum value of UINT).

The lowest-order bits of the input value are only converted to the output value.

<u>Parameter</u>	<u>Data type</u>	<u>Description</u>
IN	DINT	input value
OUT	UINT	output value

DINT_TO_USINT

Description

This type conversion function changes an input value of the data type DINT into an output value of the data type USINT.

The output values are always positive.

For example the input value -128 results in the output value 128 and the input value -1 results in the output value 255 (maximum value of USINT).

The lowest-order bits of the input value are only converted to the output value.

<u>Parameter</u>	<u>Data type</u>	<u>Description</u>
IN	DINT	input value
OUT	USINT	output value

DINT_TO_WORD

Description

This type conversion function changes an input value of the data type DINT into an output value of the data type WORD.

The lowest-order bits of the input value are only converted to the output value. For example the input value -1 results in the output value 16#FFFF.

<u>Parameter</u>	<u>Data type</u>	<u>Description</u>
IN	DINT	input value
OUT	WORD	output value



DWORD_TO_BOOL

Description

This type conversion function changes an input value of the data type DWORD into an output value of the data type BOOL.

Any input value unequal to 16#00000000 results in the output value TRUE.

<u>Parameter</u>	<u>Data type</u>	<u>Description</u>
IN	DWORD	input value
OUT	BOOL	output value

DWORD_TO_BYTE

Description

This type conversion function changes an input value of the data type DWORD into an output value of the data type BYTE.

The lowest-order bits of the input value are only converted to the output value. For example the input value 16#FFFFFF7F results in the output value 16#7F.

<u>Parameter</u>	<u>Data type</u>	<u>Description</u>
IN	DWORD	input value
OUT	BYTE	output value

DWORD_TO_DINT

Description

This type conversion function changes an input value of the data type DWORD into an output value of the data type DINT. A signed conversion is performed. For example the input value 16#80000000 results in the output value -2.147.483.648.

<u>Parameter</u>	<u>Data type</u>	<u>Description</u>
IN	DWORD	input value
OUT	DINT	output value



DWORD_TO_INT

Description

This type conversion function changes an input value of the data type DWORD into an output value of the data type DINT.

The lowest-order bits of the input value are only converted to the output value. For example the input value 16#FFFF7FFF results in the output value 32767.

If the MSB (most significant bit) of the low word is set, the input value is converted to a negative output value. For example the input value 16#00008000 results in the output value -32768.

<u>Parameter</u>	<u>Data type</u>	<u>Description</u>
IN	DWORD	input value
OUT	INT	output value

DWORD_TO_REAL

Description

This type conversion function changes an input value of the data type DWORD into an output value of the data type REAL.

The MSB (most significant bit) of the DWORD value is interpreted as sign bit. DWORD values greater than 16#7FFFFFFF result in negative values of the output data type. For example the input value 16#80000000 results in the output value -2.1474836 E+09.

<u>Parameter</u>	<u>Data type</u>	<u>Description</u>
IN	DWORD	input value
OUT	REAL	output value

DWORD_TO_SINT

Description

This type conversion function changes an input value of the data type DWORD into an output value of the data type SINT.

The lowest-order bits of the input value are only converted to the output value. For example the input value 16#FFFFFF7F results in the output value 127.

If the MSB (most significant bit) of the low byte is set, the input value is converted to a negative output value. For example the input value 16#00000080 results in the output value -128.



<u>Parameter</u>	<u>Data type</u>	<u>Description</u>
IN	DWORD	input value
OUT	SINT	output value

DWORD_TO_WORD

Description

This type conversion function changes an input value of the data type DWORD into an output value of the data type WORD.

The lowest-order bits of the input value are only converted to the output value. For example the input value 16#ABCDFFFF results in the output value 16#FFFF.

<u>Parameter</u>	<u>Data type</u>	<u>Description</u>
IN	DWORD	input value
OUT	WORD	output value

DWORD_TO_UDINT

Description

This type conversion function changes an input value of the data type DWORD into an output value of the data type UDINT.

The output values are always positive.

For example the input value 16#80000000 results in the output value 2.147.483.648.

<u>Parameter</u>	<u>Data type</u>	<u>Description</u>
IN	DWORD	input value
OUT	UDINT	output value

DWORD_TO_UINT

Description

This type conversion function changes an input value of the data type DWORD into an output value of the data type UINT.

The output values are always positive.

The lowest-order bits of the input value are only converted to the output value. For example the input value 16#00008000 results in the output value 32768.

<u>Parameter</u>	<u>Data type</u>	<u>Description</u>
IN	DWORD	input value
OUT	UINT	output value



DWORD_TO_USINT

Description

This type conversion function changes an input value of the data type DWORD into an output value of the data type USINT.

The output values are always positive.

The lowest-order bits of the input value are only converted to the output value. For example the input value 16#00000080 results in the output value 128.

<u>Parameter</u>	<u>Data type</u>	<u>Description</u>
IN	DWORD	input value
OUT	USINT	output value

INT_TO_BOOL

Description

This type conversion function changes an input value of the data type INT into an output value of the data type BOOL.

Any input value unequal to 0 results in the output value TRUE.

<u>Parameter</u>	<u>Data type</u>	<u>Description</u>
IN	INT	input value
OUT	BOOL	output value

INT_TO_BYTE

Description

This type conversion function changes an input value of the data type INT into an output value of the data type BYTE.

The lowest-order bits of the input value are only converted to the output value. For example the input value 32767 results in the output value 16#FF.

<u>Parameter</u>	<u>Data type</u>	<u>Description</u>
IN	INT	input value
OUT	BYTE	output value



INT_TO_DINT

Description

This type conversion function changes an input value of the data type INT into an output value of the data type DINT.

<u>Parameter</u>	<u>Data type</u>	<u>Description</u>
IN	INT	input value
OUT	DINT	output value

INT_TO_DWORD

Description

This type conversion function changes an input value of the data type INT into an output value of the data type DWORD.

The bit pattern of the input is converted to the lowest-order bits of the output. For example the input value 32767 results in the output value 16#00007FFF.

<u>Parameter</u>	<u>Data type</u>	<u>Description</u>
IN	INT	input value
OUT	DWORD	output value

INT_TO_REAL

Description

This type conversion function changes an input value of the data type INT into an output value of the data type REAL.

<u>Parameter</u>	<u>Data type</u>	<u>Description</u>
IN	INT	input value
OUT	REAL	output value

INT_TO_SINT

Description

This type conversion function changes an input value of the data type INT into an output value of the data type SINT.

The lowest-order bits of the input value are only converted to the output value. For example the input value 32767 results in the output value -1.

<u>Parameter</u>	<u>Data type</u>	<u>Description</u>
IN	INT	input value
OUT	SINT	output value



INT_TO_UDINT

Description

This type conversion function changes an input value of the data type INT into an output value of the data type UDINT.

The output values are always positive.

<u>Parameter</u>	<u>Data type</u>	<u>Description</u>
IN	INT	input value
OUT	UDINT	output value

INT_TO_UINT

Description

This type conversion function changes an input value of the data type INT into an output value of the data type UINT.

The output values are always positive.

<u>Parameter</u>	<u>Data type</u>	<u>Description</u>
IN	INT	input value
OUT	UINT	output value

INT_TO_USINT

Description

This type conversion function changes an input value of the data type INT into an output value of the data type USINT.

The output values are always positive.

The lowest-order bits of the input value are only converted to the output value.

<u>Parameter</u>	<u>Data type</u>	<u>Description</u>
IN	INT	input value
OUT	USINT	output value



INT_TO_WORD

Description

This type conversion function changes an input value of the data type INT into an output value of the data type WORD.

<u>Parameter</u>	<u>Data type</u>	<u>Description</u>
IN	INT	input value
OUT	WORD	output value

SINT_TO_BOOL

Description

This type conversion function changes an input value of the data type SINT into an output value of the data type BOOL.

Any input value unequal to 0 results in the output value TRUE.

<u>Parameter</u>	<u>Data type</u>	<u>Description</u>
IN	SINT	input value
OUT	BOOL	output value

SINT_TO_BYTE

Description

This type conversion function changes an input value of the data type SINT into an output value of the data type BYTE.

<u>Parameter</u>	<u>Data type</u>	<u>Description</u>
IN	INT	input value
OUT	BYTE	output value

SINT_TO_DINT

Description

This type conversion function changes an input value of the data type SINT into an output value of the data type DINT.

<u>Parameter</u>	<u>Data type</u>	<u>Description</u>
IN	SINT	input value
OUT	DINT	output value



SINT_TO_DWORD

Description

This type conversion function changes an input value of the data type SINT into an output value of the data type DWORD.

The bit pattern of the input is converted to the lowest-order bits of the output. For example the input value 127 results in the output value 16#0000007F.

<u>Parameter</u>	<u>Data type</u>	<u>Description</u>
IN	SINT	input value
OUT	DWORD	output value

SINT_TO_INT

Description

This type conversion function changes an input value of the data type SINT into an output value of the data type INT.

<u>Parameter</u>	<u>Data type</u>	<u>Description</u>
IN	SINT	input value
OUT	INT	output value

SINT_TO_REAL

Description

This type conversion function changes an input value of the data type SINT into an output value of the data type REAL.

<u>Parameter</u>	<u>Data type</u>	<u>Description</u>
IN	SINT	input value
OUT	REAL	output value

SINT_TO_UDINT

Description

This type conversion function changes an input value of the data type SINT into an output value of the data type UDINT.

The output values are always positive.

<u>Parameter</u>	<u>Data type</u>	<u>Description</u>
IN	SINT	input value
OUT	UDINT	output value



SINT_TO_UINT

Description

This type conversion function changes an input value of the data type SINT into an output value of the data type UINT.

The output values are always positive.

<u>Parameter</u>	<u>Data type</u>	<u>Description</u>
IN	SINT	input value
OUT	UINT	output value

SINT_TO_USINT

Description

This type conversion function changes an input value of the data type SINT into an output value of the data type USINT.

The output values are always positive.

<u>Parameter</u>	<u>Data type</u>	<u>Description</u>
IN	SINT	input value
OUT	USINT	output value

SINT_TO_WORD

Description

This type conversion function changes an input value of the data type SINT into an output value of the data type WORD.

The bit pattern of the input is converted to the lowest-order bits of the output. For example the input value 127 results in the output value 16#007F.

<u>Parameter</u>	<u>Data type</u>	<u>Description</u>
IN	SINT	input value
OUT	WORD	output value



TIME_TO_DINT

Description

This type conversion function changes an input value of the data type REAL into an output value of the data type DINT.

The data type TIME is interpreted as unsigned value. Following this time values greater than 2 147 483 647 ms are converted to negative values of the data type DINT. For example the input value 4294967.295 s results in the output value -1.

<u>Parameter</u>	<u>Data type</u>	<u>Description</u>
IN	TIME	input value
OUT	DINT	output value

UDINT_TO_BOOL

Description

This type conversion function changes an input value of the data type UDINT into an output value of the data type BOOL.

Any input value unequal to 0 results in the output value TRUE.

<u>Parameter</u>	<u>Data type</u>	<u>Description</u>
IN	UDINT	input value
OUT	BOOL	output value

UINT_TO_BOOL

Description

This type conversion function changes an input value of the data type UINT into an output value of the data type BOOL.

Any input value unequal to 0 results in the output value TRUE.

<u>Parameter</u>	<u>Data type</u>	<u>Description</u>
IN	UINT	input value
OUT	BOOL	output value



USINT_TO_BOOL

Description

This type conversion function changes an input value of the data type USINT into an output value of the data type BOOL.

Any input value unequal to 0 results in the output value TRUE.

<u>Parameter</u>	<u>Data type</u>	<u>Description</u>
IN	USINT	input value
OUT	BOOL	output value

UDINT_TO_BYTE

Description

This type conversion function changes an input value of the data type UDINT into an output value of the data type BYTE.

The lowest-order bits of the input value are only converted to the output value. For example the input value 254 results in the output value 16#FE.

<u>Parameter</u>	<u>Data type</u>	<u>Description</u>
IN	UDINT	input value
OUT	BYTE	output value

UINT_TO_BYTE

Description

This type conversion function changes an input value of the data type UINT into an output value of the data type BYTE.

The lowest-order bits of the input value are only converted to the output value. For example the input value 254 results in the output value 16#FE.

<u>Parameter</u>	<u>Data type</u>	<u>Description</u>
IN	UINT	input value
OUT	BYTE	output value



USINT_TO_BYTE

Description

This type conversion function changes an input value of the data type USINT into an output value of the data type BYTE.

For example the input value 64 results in the output value 16#80.

<u>Parameter</u>	<u>Data type</u>	<u>Description</u>
IN	USINT	input value
OUT	BYTE	output value

UDINT_TO_DINT

Description

This type conversion function changes an input value of the data type UDINT into an output value of the data type DINT.

If the input value is > 2.147.483.647 the output value converts to the negative range of DINT.

<u>Parameter</u>	<u>Data type</u>	<u>Description</u>
IN	UDINT	input value
OUT	DINT	output value

UINT_TO_DINT

Description

This type conversion function changes an input value of the data type UINT into an output value of the data type DINT.

<u>Parameter</u>	<u>Data type</u>	<u>Description</u>
IN	UINT	input value
OUT	DINT	output value

USINT_TO_DINT

Description

This type conversion function changes an input value of the data type USINT into an output value of the data type DINT.

<u>Parameter</u>	<u>Data type</u>	<u>Description</u>
IN	USINT	input value
OUT	DINT	output value



UDINT_TO_DWORD

Description

This type conversion function changes an input value of the data type UDINT into an output value of the data type DWORD.

<u>Parameter</u>	<u>Data type</u>	<u>Description</u>
IN	UDINT	input value
OUT	DWORD	output value

UINT_TO_DWORD

Description

This type conversion function changes an input value of the data type UINT into an output value of the data type DWORD.

<u>Parameter</u>	<u>Data type</u>	<u>Description</u>
IN	UINT	input value
OUT	DWORD	output value

USINT_TO_DWORD

Description

This type conversion function changes an input value of the data type USINT into an output value of the data type DWORD.

<u>Parameter</u>	<u>Data type</u>	<u>Description</u>
IN	USINT	input value
OUT	DWORD	output value

UDINT_TO_INT

Description

This type conversion function changes an input value of the data type UDINT into an output value of the data type INT.

The lowest-order bits of the input value are only converted to the output value. For example the input value 500 results in the output value 500.

If the input value is > 32.767 the output value converts to the negative range of INT.

<u>Parameter</u>	<u>Data type</u>	<u>Description</u>
IN	UDINT	input value
OUT	INT	output value



UINT_TO_INT

Description

This type conversion function changes an input value of the data type UINT into an output value of the data type INT.

For example the input value 500 results in the output value 500.

If the input value is > 32.767 the output value converts to the negative range of INT.

<u>Parameter</u>	<u>Data type</u>	<u>Description</u>
IN	UINT	input value
OUT	INT	output value

USINT_TO_INT

Description

This type conversion function changes an input value of the data type USINT into an output value of the data type INT.

For example the input value 100 results in the output value 100.

<u>Parameter</u>	<u>Data type</u>	<u>Description</u>
IN	USINT	input value
OUT	INT	output value

UDINT_TO_REAL

Description

This type conversion function changes an input value of the data type UDINT into an output value of the data type REAL.

<u>Parameter</u>	<u>Data type</u>	<u>Description</u>
IN	UDINT	input value
OUT	REAL	output value

UINT_TO_REAL

Description

This type conversion function changes an input value of the data type UINT into an output value of the data type REAL.

<u>Parameter</u>	<u>Data type</u>	<u>Description</u>
IN	UINT	input value
OUT	REAL	output value



USINT_TO_REAL

Description

This type conversion function changes an input value of the data type USINT into an output value of the data type REAL.

<u>Parameter</u>	<u>Data type</u>	<u>Description</u>
IN	USINT	input value
OUT	REAL	output value

UDINT_TO_SINT

Description

This type conversion function changes an input value of the data type UDINT into an output value of the data type SINT.

The lowest-order bits of the input value are only converted to the output value. For example the input value 4.294.967.295 results in the output value -1 and the input value 30 results in the output value 30.

<u>Parameter</u>	<u>Data type</u>	<u>Description</u>
IN	UDINT	input value
OUT	SINT	output value

UINT_TO_SINT

Description

This type conversion function changes an input value of the data type UINT into an output value of the data type SINT.

The lowest-order bits of the input value are only converted to the output value. For example the input value 65535 results in the output value -1 and the input value 30 results in the output value 30.

<u>Parameter</u>	<u>Data type</u>	<u>Description</u>
IN	UINT	input value
OUT	SINT	output value



USINT_TO_SINT

Description

This type conversion function changes an input value of the data type USINT into an output value of the data type SINT.

For example the input value 254 results in the output value -2 and the input value 30 results in the output value 30.

<u>Parameter</u>	<u>Data type</u>	<u>Description</u>
IN	USINT	input value
OUT	SINT	output value

UDINT_TO_UINT

Description

This type conversion function changes an input value of the data type UDINT into an output value of the data type UINT.

The lowest-order bits of the input value are only converted to the output value.

For example the input value 4.294.967.295 (maximum of UDINT) results in the output value 65535 (maximum value of UINT) and the input value 1000 results in the output value 1000.

<u>Parameter</u>	<u>Data type</u>	<u>Description</u>
IN	UDINT	input value
OUT	UINT	output value

UDINT_TO_USINT

Description

This type conversion function changes an input value of the data type UDINT into an output value of the data type USINT.

The lowest-order bits of the input value are only converted to the output value.

For example the input value 4.294.967.295 (maximum of UDINT) results in the output value 255 (maximum value of USINT) and the input value 100 results in the output value 100.

<u>Parameter</u>	<u>Data type</u>	<u>Description</u>
IN	USINT	input value
OUT	UINT	output value



UINT_TO_UDINT

Description

This type conversion function changes an input value of the data type UINT into an output value of the data type UDINT.

<u>Parameter</u>	<u>Data type</u>	<u>Description</u>
IN	UINT	input value
OUT	UDINT	output value

UINT_TO_USINT

Description

This type conversion function changes an input value of the data type UINT into an output value of the data type USINT.

The lowest-order bits of the input value are only converted to the output value. For example the input value 65535 results in the output value 255.

<u>Parameter</u>	<u>Data type</u>	<u>Description</u>
IN	UINT	input value
OUT	USINT	output value

USINT_TO_UDINT

Description

This type conversion function changes an input value of the data type USINT into an output value of the data type UDINT.

<u>Parameter</u>	<u>Data type</u>	<u>Description</u>
IN	USINT	input value
OUT	UDINT	output value

USINT_TO_UINT

Description

This type conversion function changes an input value of the data type USINT into an output value of the data type UINT.

<u>Parameter</u>	<u>Data type</u>	<u>Description</u>
IN	USINT	input value
OUT	UINT	output value



UDINT_TO_WORD

Description

This type conversion function changes an input value of the data type UDINT into an output value of the data type WORD.

The lowest-order bits of the input value are only converted to the output value. For example the input value 4.294.967.295 (maximum of UDINT) results in the output value 16#FFFF.

<u>Parameter</u>	<u>Data type</u>	<u>Description</u>
IN	UDINT	input value
OUT	WORD	output value

UINT_TO_WORD

Description

This type conversion function changes an input value of the data type UINT into an output value of the data type WORD.

<u>Parameter</u>	<u>Data type</u>	<u>Description</u>
IN	UINT	input value
OUT	WORD	output value

USINT_TO_WORD

Description

This type conversion function changes an input value of the data type USINT into an output value of the data type WORD.

<u>Parameter</u>	<u>Data type</u>	<u>Description</u>
IN	USINT	input value
OUT	WORD	output value

WORD_TO_BOOL

Description

This type conversion function changes an input value of the data type WORD into an output value of the data type BOOL.

Any input value unequal to 16#00 results in the output value TRUE.

<u>Parameter</u>	<u>Data type</u>	<u>Description</u>
IN	WORD	input value
OUT	BOOL	output value



WORD_TO_BYTE

Description

This type conversion function changes an input value of the data type WORD into an output value of the data type BYTE.

The lowest-order bits of the input value are only converted to the output value. For example the input value 16#FF7F results in the output value 16#7F.

<u>Parameter</u>	<u>Data type</u>	<u>Description</u>
IN	WORD	input value
OUT	BYTE	output value

WORD_TO_DINT

Description

This type conversion function changes an input value of the data type WORD into an output value of the data type DINT.

A signed conversion is performed. For output data types greater than the input data type a sign-conformed expansion is additionally performed. For example the input value 16#8000 results in the output value -32768.

The bit pattern of the input is converted to the lowest-order bits of the output. For example the input value 16#FFFF results in the output value -1.

<u>Parameter</u>	<u>Data type</u>	<u>Description</u>
IN	WORD	input value
OUT	DINT	output value

WORD_TO_DWORD

Description

This type conversion function changes an input value of the data type WORD into an output value of the data type DWORD.

The bit pattern of the input is converted to the lowest-order bits of the output. For example the input value 16#FFFF results in the output value 16#0000FFFF.

<u>Parameter</u>	<u>Data type</u>	<u>Description</u>
IN	WORD	input value
OUT	DWORD	output value



WORD_TO_INT

Description

This type conversion function changes an input value of the data type WORD into an output value of the data type INT.

A signed conversion is performed. For output data types greater than the input data type a sign-conformed expansion is additionally performed. For example the input value 16#8000 results in the output value -32768.

<u>Parameter</u>	<u>Data type</u>	<u>Description</u>
IN	WORD	input value
OUT	INT	output value

WORD_TO_REAL

Description

This type conversion function changes an input value of the data type WORD into an output value of the data type REAL.

The MSB (most significant bit) of the WORD value is interpreted as sign bit. WORD values greater than 16#7FFF result in negative values of the output data type. For example the input value 16#8000 results in the output value -3.2768000 E+04.

The bit pattern of the input is converted to the lowest-order bits of the output. For example the input value 16#FFFF results in the output value -1.0000000 E+00.

<u>Parameter</u>	<u>Data type</u>	<u>Description</u>
IN	WORD	input value
OUT	REAL	output value

WORD_TO_SINT

Description

This type conversion function changes an input value of the data type WORD into an output value of the data type SINT.

The lowest-order bits of the input value are only converted to the output value. For example the input value 16#FF7F results in the output value 127.

If the MSB (most significant bit) of the low byte is set, the input value is converted to a negative output value. For example the input value 16#0080 results in the output value -128.

<u>Parameter</u>	<u>Data type</u>	<u>Description</u>
IN	WORD	input value
OUT	SINT	output value



WORD_TO_UDINT

Description

This type conversion function changes an input value of the data type WORD into an output value of the data type UDINT.

The bit pattern of the input is converted to the lowest-order bits of the output. For example the input value 16#FFFF results in the output value 65535.

<u>Parameter</u>	<u>Data type</u>	<u>Description</u>
IN	WORD	input value
OUT	UDINT	output value

WORD_TO_UINT

Description

This type conversion function changes an input value of the data type WORD into an output value of the data type UINT.

<u>Parameter</u>	<u>Data type</u>	<u>Description</u>
IN	WORD	input value
OUT	UINT	output value

WORD_TO_USINT

Description

This type conversion function changes an input value of the data type WORD into an output value of the data type USINT.

The lowest-order bits of the input value are only converted to the output value. For example the input value 16#FFFF results in the output value 255 (maximum of USINT).

<u>Parameter</u>	<u>Data type</u>	<u>Description</u>
IN	WORD	input value
OUT	USINT	output value