

L-force Controls

Software Manual

Function library "LenzeConversionBox"

for Lenze software »Drive PLC Developer Studio« and »PLC Designer«

Contents

1	About this documentation	3
1.1	Conventions used	4
1.2	Definition of notes used	5
2	System requirements	6
3	Function blocks	7
3.1	L_cbByteArrayToByte - Convert array of byte into 8 individual bytes	8
3.2	L_cbByteToByteArray - Convert 8 individual bytes into an array of byte	9
3.3	L_cbByteArrayToInt - Convert array of byte into 4 individual integers	10
3.4	L_cbIntToByteArray - Convert 4 individual integers into an array of byte	11
4	Index	12
	Your opinion is important to us	13

1 About this documentation

This documentation describes the function blocks contained in the "LenzeConversionBox" function library.

Functional survey

- ▶ Conversion of 8 individual bytes into an 8 byte array
- ▶ Conversion of an 8 byte array into 8 individual bytes
- ▶ Conversion of 4 individual integers into an 8 byte array
- ▶ Conversion of an 8 byte array into 4 individual integers

Validity information

The information given in this documentation is valid for the following function library:

Function library	From version
LenzeConversionBoxVxx.lib	V1.0

Document history

Version			Description
1.0	07/2011	TD05	First edition for LenzeConversionBoxV10.lib



Tip!

The "LenzeConversionBoxVxx.lib" library can be queried via the following global constants:

- *C_wLenzeConversionBoxVersionER*: Enabled major version
- *C_wLenzeConversionBoxVersionEL*: Enabled minor version
- *C_wLenzeConversionBoxVersionIR*: Internal minor version
- *C_wLenzeConversionBoxVersionBN*: Internal build number

1.1 Conventions used

This documentation uses the following conventions to distinguish between different types of information:

Type of information	Writing	Examples/notes
Numbers		
Decimal separator	Point	The decimal point is always used. Example: 1234.56
Text		
Program name	» «	The Lenze PC software »PLC Designer«...
Window	<i>Italics</i>	The <i>Message</i> window ... / The <i>Options</i> dialog box...
Variable identifier		By setting <i>bEnable</i> to TRUE...
Control element	Bold	The OK button... / The copy command... / The Properties tab... / The Name input field...
Sequence of menu commands		If several commands must be used in sequence to carry out a function, then the individual commands are separated by an arrow: Select File → Open to...
Shortcut	< Bold >	Press < F1 > to open the online help. If a key combination is required for a command, a "+" is inserted between the key identifiers: Use < Shift >+< ESC >...
Program code	Courier	IF var1 < var2 THEN a = a + 1 END IF
Keyword	Courier bold	
Hyperlink	<u>Underlined</u>	Optically highlighted reference to another topic. It is activated with a mouse click in this online documentation.
Symbols		
Page reference	 4	Optically highlighted reference to another page. It is activated with a mouse click in this online documentation.
Step-by-step instructions		Step-by-step instructions are indicated by a pictograph.

1.2 Definition of notes used

The following signal words and symbols are used in this documentation to indicate dangers and important information:

Safety instructions

Layout of the safety instructions:



Danger!

(characterises the type and severity of danger)

Note

(describes the danger and gives information about how to prevent dangerous situations)

Pictograph	Signal word	Meaning
	Danger!	Danger of personal injury through dangerous electrical voltage Reference to an imminent danger that may result in death or serious personal injury if the corresponding measures are not taken.
	Danger!	Danger of personal injury through a general source of danger Reference to an imminent danger that may result in death or serious personal injury if the corresponding measures are not taken.
	Stop!	Danger of property damage Reference to a possible danger that may result in property damage if the corresponding measures are not taken.

Application notes

Pictograph	Signal word	Meaning
	Note!	Important note to ensure trouble-free operation
	Tip!	Useful tip for simple handling
		Reference to another document

2 System requirements

Software

The function library can be used with the following Lenze software:

Product	Type designation	Version
Drive PLC Developer Studio	ESP-DDS2-x	2.x
PLC Designer V2		2.2 or higher

Supported target systems

Product range	Type designation	Hardware version	From software version
Drive PLC	EPL-10200	1A or higher	2.x
9300 Servo PLC	EVS93xx-EI EVS93xx-ET	2M or higher	2.x
ECS	ECSxA	1B or higher	7.x
9400 ServoPLC	E94AxPExxxx	2A or higher	2.x

3 Function blocks

Function block	Function
L_cbByteArrayToByte	Convert array of byte into 8 individual bytes
L_cbByteToByteArray	Convert 8 individual bytes into an array of byte
L_cbByteArrayToInt	Convert array of byte into 4 individual integers
L_cbIntToByteArray	Convert 4 individual integers into an array of byte

Function library "LenzeConversionBox"

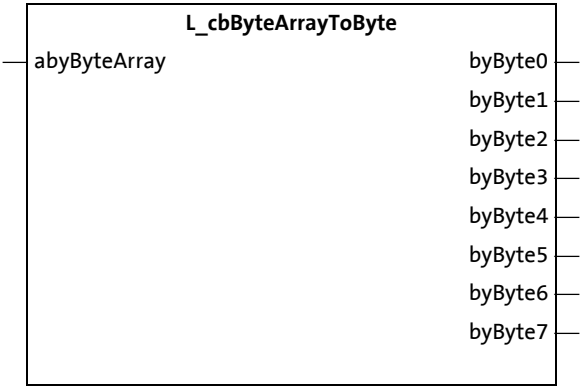
Function blocks

L_cbByteArrayToByte - Convert array of byte into 8 individual bytes

3.1 L_cbByteArrayToByte - Convert array of byte into 8 individual bytes

Call possible in:	☒ Cyclic task	☒ Time-controlled task (INTERVAL)
	☒ Interrupt task	☒ Event-controlled task (EVENT)

This FB converts an ARRAY OF BYTE consisting of 8 elements from the BYTE data type into 8 individual bytes.



Inputs

Identifier/data type	Information/possible settings
abyByteArray ARRAY[1..8] OF BYTE	Array with 8 elements from BYTE data type

Outputs

Identifier/data type	Value/meaning
byByte0 ... byByte7 BYTE	Element 1 ... 8 of the <i>abyByteArray</i> input

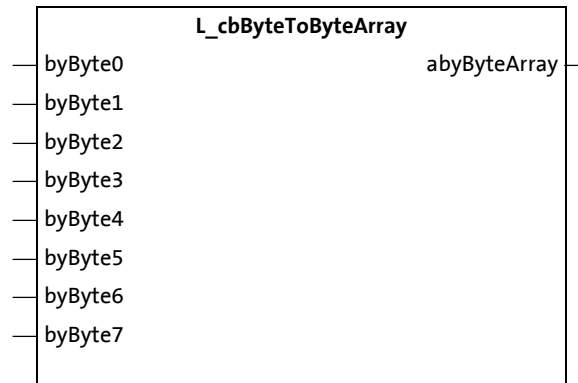
3.2 L_cbByteToByteArray - Convert 8 individual bytes into an array of byte

Call possible in:

- ☒ Cyclic task
- ☒ Interrupt task

- ☒ Time-controlled task (INTERVAL)
- ☒ Event-controlled task (EVENT)

This FB converts 8 individual bytes into an ARRAY OF BYTE consisting of 8 elements of the BYTE data type.



Inputs

Identifier/data type	Information/possible settings
byByte0 ... byByte7 BYTE	Element 1 ... 8 of the <i>abyByteArray</i> output

Outputs

Identifier/data type	Value/meaning
abyByteArray ARRAY[1..8] OF BYTE	Array with 8 elements from BYTE data type

Function library "LenzeConversionBox"

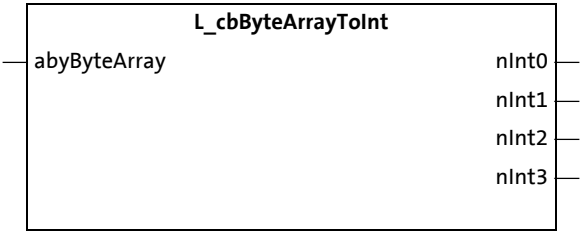
Function blocks

L_cbByteArrayToInt - Convert array of byte into 4 individual integers

3.3 L_cbByteArrayToInt - Convert array of byte into 4 individual integers

Call possible in:	☒ Cyclic task	☒ Time-controlled task (INTERVAL)
	☒ Interrupt task	☒ Event-controlled task (EVENT)

This FB converts an ARRAY OF BYTE consisting of 8 elements of the BYTE data type into 4 individual integers. The element 1 of the array is the lower-order byte of the *nInt0* output and the element 2 is the higher-order byte of the *nInt0* output, etc.



Inputs

Identifier/data type	Information/possible settings
abyByteArray ARRAY[1..8] OF BYTE	Array with 8 elements from BYTE data type

Outputs

Identifier/data type	Value/meaning
nInt0 INT	Element 1+2 of the <i>abyByteArray</i> input
nInt1 INT	Element 3+4 of the <i>abyByteArray</i> input
nInt2 INT	Element 5+6 of the <i>abyByteArray</i> input
nInt3 INT	Element 7+8 of the <i>abyByteArray</i> input

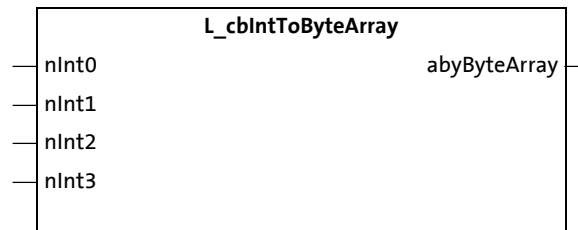
3.4 L_cbIntToByteArray - Convert 4 individual integers into an array of byte

Call possible in:

- ☒ Cyclic task
- ☒ Interrupt task

- ☒ Time-controlled task (INTERVAL)
- ☒ Event-controlled task (EVENT)

This FB converts 4 individual integers into an ARRAY OF BYTE consisting of 8 elements of the BYTE data type. The element 1 of the array is the lower-order byte of the *nInt0* input and the element 2 is the higher-order byte of the *nInt0* input, etc.



Inputs

Identifier/data type	Information/possible settings
nInt0 INT	Element 1+2 of the <i>abyByteArray</i> output
nInt1 INT	Element 3+4 of the <i>abyByteArray</i> output
nInt2 INT	Element 5+6 of the <i>abyByteArray</i> output
nInt3 INT	Element 7+8 of the <i>abyByteArray</i> output

Outputs

Identifier/data type	Value/meaning
abyByteArray ARRAY[1..8] OF BYTE	Array with 8 elements from BYTE data type

4 Index

A

Application notes [5](#)

D

Document history [3](#)

E

E-mail to Lenze [13](#)

F

Feedback to Lenze [13](#)

L

L_cbByteArrayToByte [8](#)

L_cbByteArrayToInt [10](#)

L_cbByteToByteArray [9](#)

L_cbIntToByteArray [11](#)

Layout of the safety information [5](#)

Layout of the safety instructions [5](#)

S

Safety instructions [5](#)

V

Validity information [3](#)

Version identifier of the function library [3](#)



Your opinion is important to us

These instructions were created to the best of our knowledge and belief to give you the best possible support for handling our product.

Perhaps we have not always succeeded in achieving this goal. If you notice this, please send us your suggestions and criticism in a short e-mail to:

feedback-docu@Lenze.de

Thank you for your support.

Your Lenze documentation team



© 07/2011



Lenze Automation GmbH
Hans-Lenze-Str. 1
D-31855 Aerzen
Germany



+49 (0)51 54 / 82-0



+49 (0)51 54 / 82-28 00



Lenze@Lenze.de



www.Lenze.com

Service Lenze Service GmbH
Breslauer Straße 3
D-32699 Extertal
Germany



00 80 00 / 24 4 68 77 (24 h helpline)



+49 (0)51 54 / 82-11 12



Service@Lenze.de

EDSPLCLIB02 ■ 13384446 ■ EN 1.0 ■ TD05

10 9 8 7 6 5 4 3 2 1