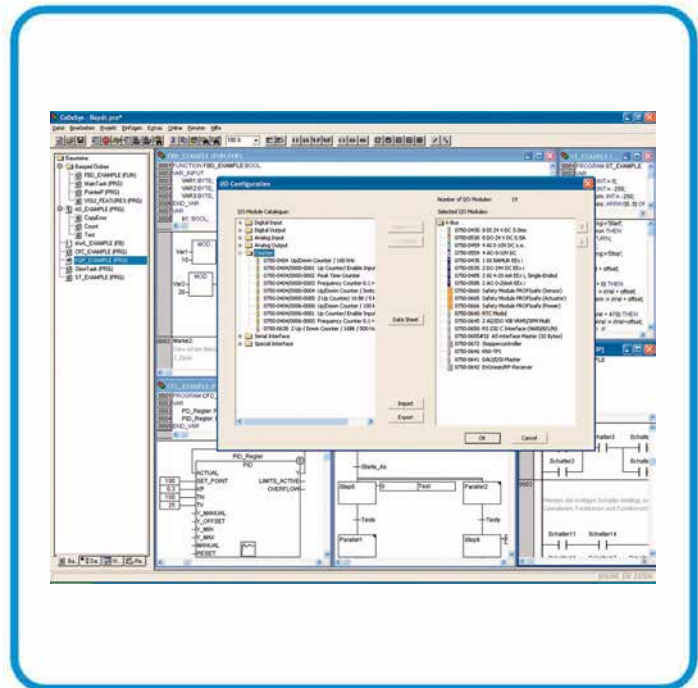


Library Description

AUTOMATION



WAGO Software CODESYS Library WagoLibLed.lib

for control and querying of user LEDs and querying of status LEDs

Version 1.0.0

WAGO®
INNOVATIVE CONNECTIONS

© 2014 by WAGO Kontakttechnik GmbH & Co. KG
All rights reserved.

WAGO Kontakttechnik GmbH & Co. KG

Hansastraße 27
D-32423 Minden

Phone: +49 (0) 571/8 87 – 0
Fax: +49 (0) 571/8 87 – 1 69

E-Mail: info@wago.com

Web: <http://www.wago.com>

Technical Support

Phone: +49 (0) 571/8 87 – 5 55
Fax: +49 (0) 571/8 87 – 85 55

E-Mail: support@wago.com

Every conceivable measure has been taken to ensure the accuracy and completeness of this documentation. However, as errors can never be fully excluded, we always appreciate any information or suggestions for improving the documentation.

E-Mail: documentation@wago.com

We wish to point out that the software and hardware terms as well as the trademarks of companies used and/or mentioned in the present manual are generally protected by trademark or patent.

Table of Contents

1	Notes about this Documentation.....	4
1.1	Copyright.....	4
1.2	Symbols.....	5
1.3	Number Notation.....	7
1.4	Font Conventions	7
2	WagoLibLed.lib	8
2.1	LED_SET_STATIC	9
2.2	LED_SET_BLINK.....	10
2.3	LED_SET_FLASH	12
2.4	LED_SET_ERROR.....	14
2.5	LED_RESET_ERROR	15
2.6	LED_RESET_ALL_ERRORS	16
2.7	LED_GET_STATE.....	17
2.8	LED_GET_STATE_ASYNC	18
2.9	LED_LedColor.....	20
2.10	LED_LedState.....	21
2.11	LED_STATE.....	22
	List of Tables.....	23

1 Notes about this Documentation

1.1 Copyright

This Manual, including all figures and illustrations, is copyright-protected. Any further use of this Manual by third parties that violate pertinent copyright provisions is prohibited. Reproduction, translation, electronic and phototechnical filing/archiving (e.g., photocopying) as well as any amendments require the written consent of WAGO Kontakttechnik GmbH & Co. KG, Minden, Germany. Non-observance will involve the right to assert damage claims.

1.2 Symbols

 **DANGER**

Personal Injury!

Indicates a high-risk, imminently hazardous situation which, if not avoided, will result in death or serious injury.

 **DANGER**

Personal Injury Caused by Electric Current!

Indicates a high-risk, imminently hazardous situation which, if not avoided, will result in death or serious injury.

 **WARNING**

Personal Injury!

Indicates a moderate-risk, potentially hazardous situation which, if not avoided, could result in death or serious injury.

 **CAUTION**

Personal Injury!

Indicates a low-risk, potentially hazardous situation which, if not avoided, may result in minor or moderate injury.

NOTICE

Damage to Property!

Indicates a potentially hazardous situation which, if not avoided, may result in damage to property.

NOTICE

Damage to Property Caused by Electrostatic Discharge (ESD)!

Indicates a potentially hazardous situation which, if not avoided, may result in damage to property.

Note

Important Note!

Indicates a potential malfunction which, if not avoided, however, will not result in damage to property.



Information

Additional Information:

Refers to additional information which is not an integral part of this documentation (e.g., the Internet).

1.3 Number Notation

Table 1: Number notation

Number code	Example	Note
Decimal	100	Normal notation
Hexadecimal	0x64	C notation
Binary	'100' '0110.0100'	In quotation marks, nibble separated with dots (.)

1.4 Font Conventions

Table 2: Font conventions

Font type	Indicates
<i>italic</i>	Names of paths and data files are marked in italic-type. e.g.: <i>C:\Programme\WAGO-I/O-CHECK</i>
Menu	Menu items are marked in bold letters. e.g.: Save
>	A greater-than sign between two names means the selection of a menu item from a menu. e.g.: File > New
Input	Designation of input or optional fields are marked in bold letters, e.g.: Start of measurement range
“Value”	Input or selective values are marked in inverted commas. e.g.: Enter the value “4 mA” under Start of measurement range .
[Button]	Pushbuttons in dialog boxes are marked with bold letters in square brackets. e.g.: [Input]
[Key]	Keys are marked with bold letters in square brackets. e.g.: [F5]

2 **WagoLibLed.lib**

This library contains function blocks that can be used for control and querying of the user LEDs and for querying of the status LEDs. Depending on the device design, not all of the user or status LEDs may be provided.

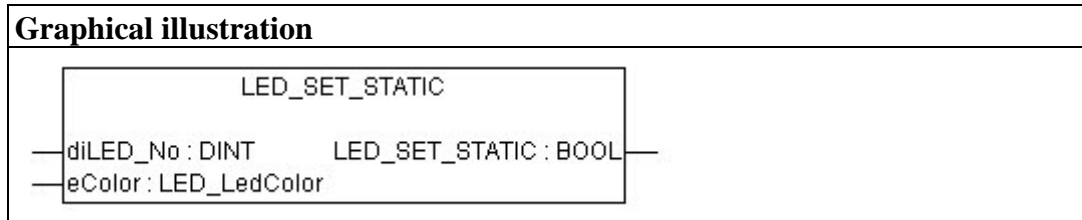
2.1 LED_SET_STATIC

This function sets the user LED to be controlled to a status in which it continuously has a defined color value.

Category	Switching of an LED
Name	LED_SET_STATIC
Type	Function
Name of library	WagoLibLed.lib
Required libraries	
Applicable to	750-820x

Input parameter		
Name	Data type	Description
diLED_No	DINT	Number of the LED to be controlled
eColor	LED_LedColor	Color in which the LED to be controlled is to light up.

Output parameter		
Name	Data type	Description
LED_SET_STATIC	BOOL	When successful, the function yields TRUE.



Description
This function is used to set the user LED to a status in which it remains lit in the defined color.

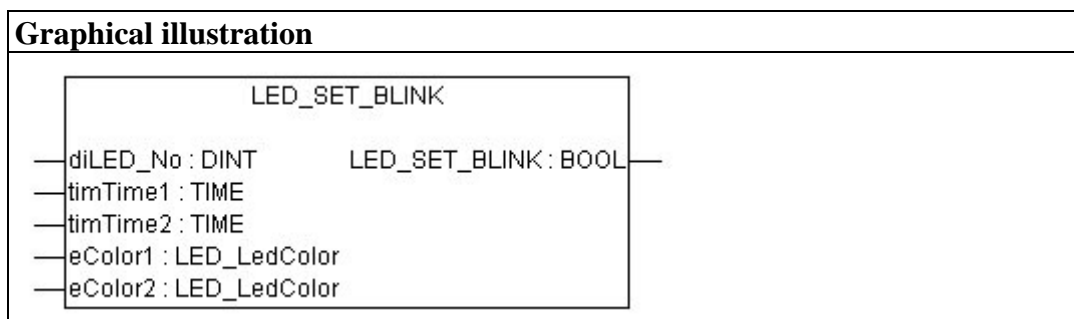
2.2 LED_SET_BLINK

This function sets the user LED to be controlled in a status in which it flashes at a defined cycle.

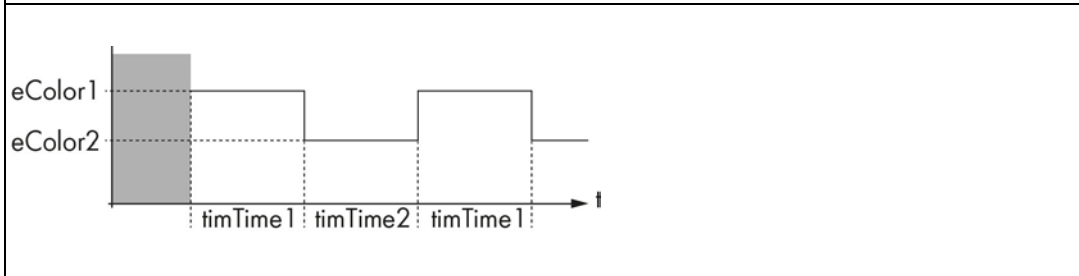
Category	Switching of an LED
Name	LED_SET_BLINK
Type	Function
Name of library	WagoLibLed.lib
Required libraries	
Applicable to	750-820x

Input parameter		
Name	Data type	Description
diLED_No	DINT	Number of the LED to be controlled
timTime1	TIME	Time in which the LED to be controlled is to light up in the defined color "eColor1".
timTime2	TIME	Time in which the LED to be controlled is to light up in the defined color "eColor2".
eColor1	LED_LedColor	Color in which the LED to be controlled is to light up over the defined time "timTime1".
eColor2	LED_LedColor	Color in which the LED to be controlled is to light up over the defined time "timTime2".

Output parameter		
Name	Data type	Description
LED_SET_BLINK	BOOL	When successful, the function yields TRUE.



Time referenced behavior



Description

This function is used to set the user LED to a state in which it lights up in two different colors alternately for freely definable time periods. This general flashing status cannot be applied to an LED that is currently flashing an error code.

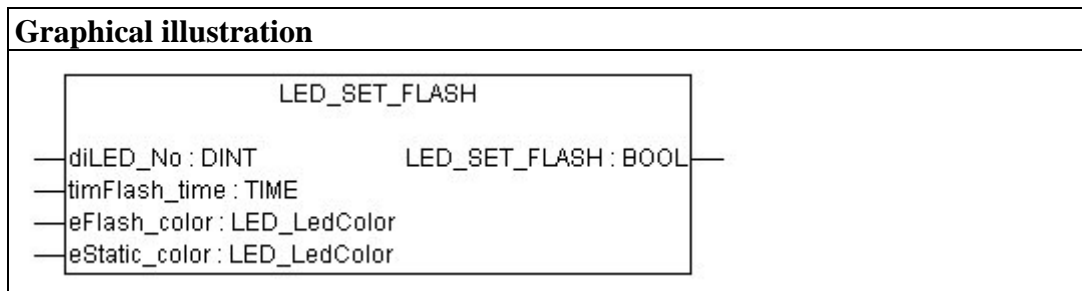
2.3 LED_SET_FLASH

This function sets the user LED to be controlled to a state in which it lights up one time and then assumes a static state.

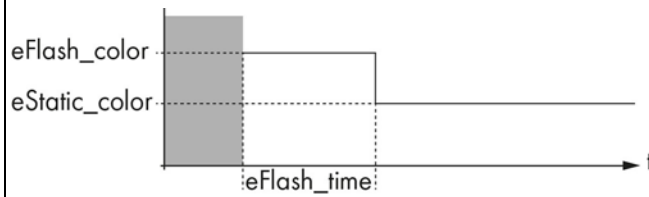
Category	Switching of an LED
Name	LED_SET_FLASH
Type	Function
Name of library	WagoLibLed.lib
Required libraries	
Applicable to	750-820x

Input parameter		
Name	Data type	Description
diLED_No	DINT	Number of the LED to be controlled
timFlash_time	TIME	Time during which the LED to be controlled is to light up with the color "eFlash_color".
eFlash_color	LED_LedColor	Color that the LED to be controlled is to light up during the "timFlash_time" time.
eStatic_color	LED_LedColor	Color that the LED to be controlled is to light up when the "timFlash_time" time expires.

Output parameter		
Name	Data type	Description
LED_SET_BLINK	BOOL	When successful, the function yields TRUE.



Time referenced behavior



Description

This function is used to set the user LED to be controlled to a freely definable color for a freely definable time, after which it then switches a static state.

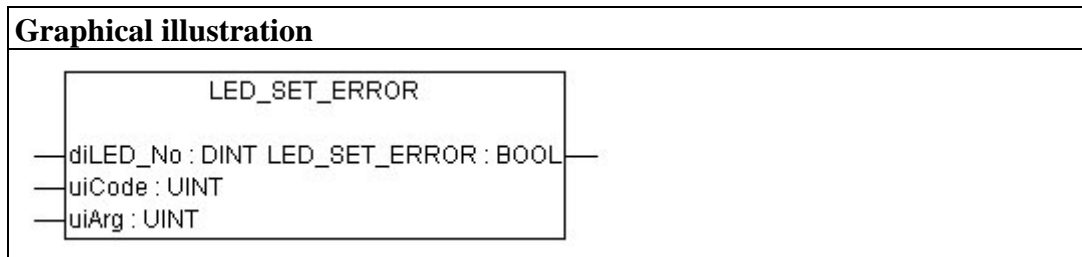
2.4 LED_SET_ERROR

This function sets the user LED to be controlled to a state in which it flashes an error code.

Category	Switching of an LED
Name	LED_SET_ERROR
Type	Function
Name of library	WagoLibLed.lib
Required Libraries	
Applicable to	750-820x

Input parameter		
Name	Data type	Description
diLED_No	DINT	Number of the LED to be controlled
uiCode	UINT	Error code. First flashing argument
uiArg	UINT	Error argument. Second flashing argument

Output parameter		
Name	Data type	Description
LED_SET_ERROR	BOOL	When successful, the function yields TRUE.



Description
<p>This function is used to set the user LED to be controlled to a state in which it flashes an error code. This flashing sequence is then identical to the flashing sequences used for system errors.</p> <p>If the intended LED is already in an error code flashing state, the specified error code is put into a queue until the currently flashing error code is canceled.</p> <p>Each error code may only be in the queue one time!</p> <p>This flashing status cannot be applied to an LED that currently has a cyclic flashing status.</p>

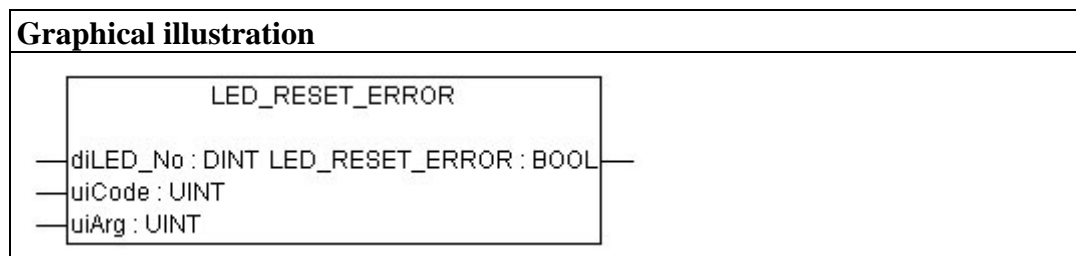
2.5 LED_RESET_ERROR

This function resets a defined error code for a user LED.

Category	Switching of an LED
Name	LED_RESET_ERROR
Type	Function
Name of library	WagoLibLed.lib
Required libraries	
Applicable to	750-820x

Input parameter		
Name	Data type	Description
diLED_No	DINT	Number of the LED to be controlled
uiCode	UINT	Error code. First flashing argument
uiArg	UINT	Error argument. Second flashing argument

Output parameter		
Name	Data type	Description
LED_RESET_ERROR	BOOL	When successful, the function yields TRUE.



Description
<p>This function cancels an error code that is currently being flashed by a user LED, or which is currently in the queue. The LED is then deactivated, provided no other error codes have been set for it.</p>

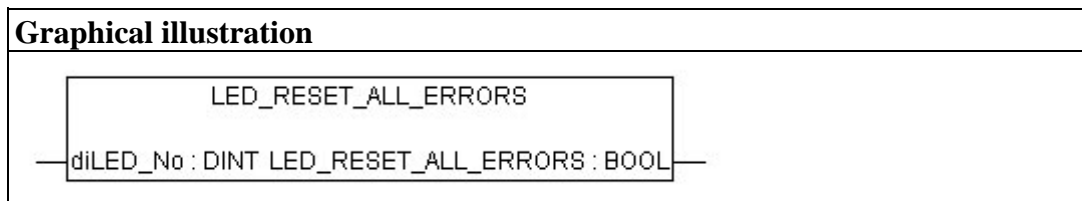
2.6 LED_RESET_ALL_ERRORS

This function resets all error codes for a user LED.

Category	Switching of an LED
Name	LED_RESET_ALL_ERRORS
Type	Function
Name of library	WagoLibLed.lib
Required libraries	
Applicable to	750-820x

Input parameter		
Name	Data type	Description
diLED_No	DINT	Number of the LED to be controlled

Output parameter		
Name	Data type	Description
LED_RESET_ALL_ERRORS	BOOL	When successful, the function yields TRUE.



Description
<p>This function cancels the error code currently being indicated, along with all error codes currently in the queue for a user LED. The LED is then deactivated.</p>

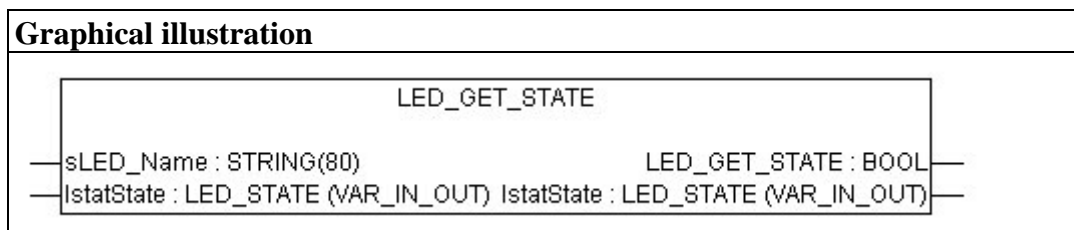
2.7 LED_GET_STATE

This function queries the current status of an LED.

Category	Querying of LEDs
Name	LED_GET_STATE
Type	Function
Name of library	WagoLibLed.lib
Required libraries	
Applicable to	750-820x

Input parameter		
Name	Data type	Description
sLED_Name	STRING(31)	Name of LED to be queried

Output parameter		
Name	Data type	Description
IstatState	LED_STATE	Returns a structure in which the current status of the LED is saved.



Description
<p>This function contains the name of an LED as an input parameter (Un, SYS, IO etc.).</p> <p>This function yields a structure in which the current status and additional parameters for the LED are stored.</p>

2.8 LED_GET_STATE_ASYNC

This function block queries the current status of an LED without blocking the ongoing process.

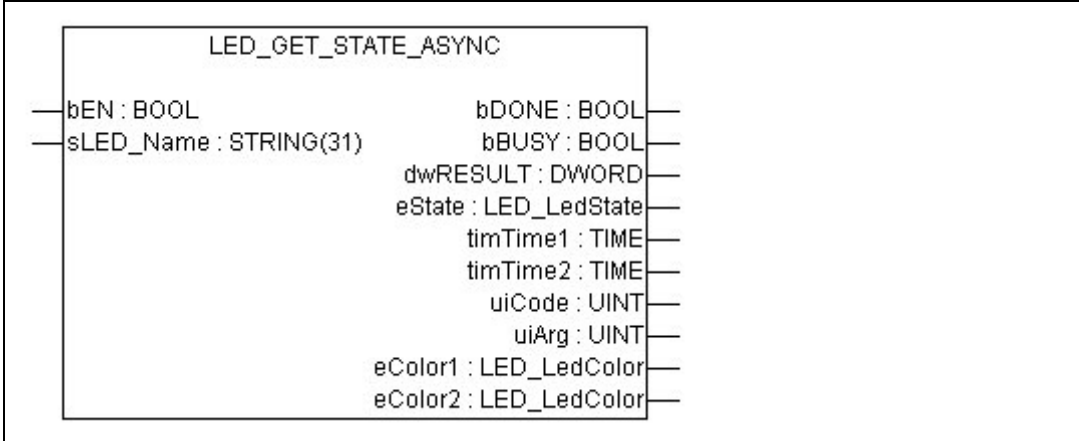
Category	Querying of LEDs
Name	LED_GET_STATE_ASYNC
Type	Function block
Name of library	WagoLibLed.lib
Required libraries	
Applicable to	750-820x

Input parameter		
Name	Data type	Description
bEN	BOOL	Activation of the function block
sLED_Name	String(31)	Name of LED to be queried

Output parameter		
Name	Data type	Description
bDONE	BOOL	Execution confirmation
bBUSY	BOOL	Module is busy
dwRESULT	DWORD	Result of operation
eState	LED_LedState	Current status of the LED
timTime1	TIME	Time period 1
timTime2	TIME	Time period 2
uiCode	UINT	Error code
uiArg	UINT	Error Argument
eColor1	LED_LedColor	Color 1
eColor2	LED_LedColor	Color 2

Variable declaration
bOldEnable:BOOL; dimqd:DINT;

Graphical illustration



Description

This function block is a non-blocking version of the LED_GET_STATE function. As querying of the LED status by the LED server may required a relatively long time, this function block provides the option for asynchronous querying of the LED in order not to impair continued execution of the rest of the program.

2.9 LED_LedColor

This data type defines a list of colors for the LED.

Category	LED description types
Name	LED_LedColor
Type	Data type (list)
Name of library	WagoLibLed.lib
Applicable to	750-820x
Structure	<pre> TYPE LED_LedColor: (LED_COLOR_OFF := 0, LED_COLOR_GREEN := 1, LED_COLOR_RED := 2, LED_COLOR_YELLOW := 3); END_TYPE </pre>

Element	Value	Description
LED_COLOR_OFF	0	The LED does not light up.
LED_COLOR_GRN	1	LED lights up green.
LED_COLOR_RED	2	LED lights up red.
LED_COLOR_YLW	3	LED lights up yellow (or orange).

Description
This list type described the colors that an LED can have.

2.10 LED_LedState

This data type defines a list of states that an LED can have.

Category	LED description types
Name	LED_LedState
Type	Data type (list)
Name of library	WagoLibLed.lib
Applicable to	750-7202, 750-820x
Structure	<pre> TYPE LED_LedState: (LED_FAIL, LED_STATE_OFF, LED_STATE_STATIC, LED_STATE_BLINK, LED_STATE_ERROR, LED_STATE_FLASH); END_TYPE </pre>

Element	Value	Description
LED_FAIL	0	General error
LED_STATE_OFF	1	LED remains unlit.
LED_STATE_STATIC	2	LED is lit.
LED_STATE_BLINK	3	LED flashing.
LED_STATE_ERROR	4	LED flashing an error code.
LED_STATE_FLASH	5	LED flashes one time.

Description
This list type describes the statuses that can be assumed by an LED. LED_FAIL is set when an error occurs during reading of the LED, or when the LED has not yet been read.

2.11 LED_STATE

This data type defines the structure which describes the status of an LED.

Category	LED description types
Name	LED_STATE
Type	Data type (structure)
Name of library	WagoLibLed.lib
Applicable to	750-7202, 750-820x
Structure	<pre> TYPE LED_STATE: STRUCT eState:LED_LedState; timTime1:TIME; timTime2:TIME; uiCode:UINT; uiArg:UINT; eColor1:LED_LedColor; eColor2:LED_LedColor; END_STRUCT END_TYPE </pre>

Element	Value	Description
eState	LED_STATE_OFF	Status
timTime1	0	First time value
timTime2	0	Second time value
uiCode	0	Error code
uiArg	0	Error Argument
eColor1	LED_COLOR_OFF	First color value
eColor2	LED_COLOR_OFF	Second color value

Description						
This structure is used for a full description of an LED. The meaning of the other fields varies according to the "eState" element:						
eState	OFF	STATIC	FLASH	FLASH	ERROR	FAIL
timTime1	(invalid)	(invalid)	Flashing time 1	Flash time	(invalid)	(invalid)
timTime2	(invalid)	(invalid)	Flashing time 2	(invalid)	(invalid)	(invalid)
uiCode	(invalid)	(invalid)	(invalid)	(invalid)	Error code	(invalid)
uiArg	(invalid)	(invalid)	(invalid)	(invalid)	Error argument:	(invalid)
eColor1	OFF	Static color	Flashing color 1	Flash color	(invalid)	(invalid)
eColor2	(invalid)	(invalid)	Flashing color 2	Static color	(invalid)	(invalid)

List of Tables

Table 1: Number notation..... 7
Table 2: Font conventions 7

WAGO Kontakttechnik GmbH & Co. KG
Postfach 2880 • D-32385 Minden
Hansastraße 27 • D-32423 Minden
Phone: +49/5 71/8 87 – 0
Fax: +49/5 71/8 87 – 1 69
E-Mail: info@wago.com
Internet: <http://www.wago.com>

