PL 221

S-DIAS Pixel LED Module

Instruction Manual

Date of creation: 13.03.2018 | Version date: 26.07.2023 | Article number: 20-018-221-E

Publisher: SIGMATEK GmbH & Co KG
A-5112 Lamprechtshausen
Tel.: +43/6274/4321
Fax: +43/6274/4321-18
Email: office@sigmatek.at
WWW.SIGMATEK-AUTOMATION.COM

Translation of the Original Instructions

S-DIAS Pixel LED Module PL 221

with control for 2x RGB pixel LED strips and 2x 24 V PWM LED strips

With the S-DIAS Pixel LED module PL 221, two pixel LED strips with a maximum of 512 pixel RGB LEDs each and two +24 V PWM dimmable white LED strips can be controlled.

The +5 V supply for the pixel LEDs is provided an internal switching regulator from the externally powered +24 V supply. The +5 V supply is protected against short circuit and overload.

The +24 V supply for the +24 V PWM dimmable white LED strips is provided by the external +24 V supply. The outputs are protected against short circuit and overload.

The external +24 V supply is monitored for undervoltage.

Contents

1 Introduction

1.1 Target Group/Purpose of this Operating Manual

This operating manual contains all information required for the operation of the product.

This operating manual is intended for:

- Project planners
- Technicians
- Commissioning engineers
- Machine operators
- Maintenance/test technicians

General knowledge of automation technology is required.

Further help and training information, as well as the appropriate accessories can be found on our website www.sigmatek-automation.com.

Our support team is happily available to answer your questions.
Please see our website for our hotline number and business hours.

1.2 Important Reference Documentation

This and additional documents can be downloaded from our website or obtained through support.

1.3 Contents of Delivery

1x PL 221

2 Basic Safety Directives

2.1 Symbols Used

The following symbols are used in the operator documentation for warning and danger messages, as well as informational notes:

DANGER

Danger indicates that death or serious injury **will occur**, if the specified measures are not taken.

⇨ To avoid death or serious injuries, observe all guidelines.

Danger indique une situation dangereuse qui, faute de prendre les mesures adéquates, **entraînera** des blessures graves, voire mortelles.

⇨ Respectez toutes les consignes pour éviter des blessures graves, voire mortelles.

WARNING

Warning indicates that death or serious injury **can** occur, if the specified measures are not taken.

⇨ To avoid death or serious injuries, observe all guidelines.

Avertissement d'une situation dangereuse qui, faute de prendre les mesures adéquates, **entraînera** des blessures graves, voire mortelles.

⇨ Respectez toutes les consignes pour éviter des blessures graves, voire mortelles.

CAUTION

Caution indicates that moderate to slight injury **can** occur, if the specified measures are not taken.

⇨ To avoid moderate to slight injuries, observe all guidelines.

Attention indique une situation dangereuse qui, faute de prendre les mesures adéquates, **peut** entraîner des blessures assez graves ou légères.

⇨ Respectez toutes les consignes pour éviter des blessures graves, voire mortelles.

INFORMATION

Information

⇨ Provides important information on the product, handling or relevant sections of the documentation, which require attention.

2.2 Disclaimer

INFORMATION

The contents of this operating manual were prepared with the greatest care. However, deviations cannot be ruled out. This operating manual is regularly checked and required corrections are included in the subsequent versions. The machine manufacturer is responsible for the proper assembly, as well as device configuration. The machine operator is responsible for safe handling, as well as proper operation.

The current operating manual can be found on our website. If necessary, contact our support.

Subject to technical changes, which improve the performance of the devices. The following operating manual is purely a product description. It does not guarantee properties under the warranty.

Please thoroughly read the corresponding documents and this operating manual before handling a product.

SIGMATEK GmbH & Co KG is not liable for damages caused through, non-compliance with these instructions or applicable regulations.

2.3 General Safety Directives

The Safety Directives in the other sections of this operating manual must be observed. These instructions are visually emphasized by symbols.

INFORMATION

According to EU Directives, the operating manual is a component of a product.

This operating manual must therefore be accessible in the vicinity of the machine since it contains important instructions.

This operating manual should be included in the sale, rental or transfer of the product, or its online availability indicated.

Regarding the requirements for Safety and health connected to the use of machines, the manufacturer must perform a risk assessment in accordance with machine directives 2006/42/EG before introducing a machine to the market.

Operate the unit with devices and accessories approved by SIGMATEK only.

CAUTION

Handle the device with care and do not drop or let fall.

Prevent foreign bodies and fluids from entering the device.

The device must not be opened!

Manipulez l'appareil avec précaution et ne le laissez pas tomber.

Empêchez les corps étrangers et les liquides de pénétrer dans l'appareil.

L'appareil ne doit pas être ouvert!

If the device does not function as intended or has damage that could pose a danger, it must be replaced!

En cas de fonctionnement non conforme ou de dommages pouvant entraîner des risques, l'appareil doit être remplacé!

The module complies with EN 61131-2.

In combination with a facility, the system integrator must comply with EN 60204-1 standards.

For your own safety and that of others, compliance with the environmental conditions is essential.

Le module est conforme à la norme EN 61131-2.

En combinaison avec une équipement, l'intégrateur de système doit respecter la norme EN 60204-1.

Pour votre propre sécurité et celle des autres, le respect des conditions environnementales est essential.

2.4 Software/Training

The application is created with the software LASAL CLASS 2 and LASAL SCREEN Editor.

Training for the LASAL development environment, with which the product can be configured, is provided. Information on our training schedule can be found on our website.

3 Standards and Directives

3.1 Directives

The product was constructed in compliance with the following European Union directives and tested for conformity.

3.1.1 EU Conformity Declaration

CE

EU Declaration of Conformity

The product PL 221 conforms to the following European directives:

- **2014/35/EU** Low-voltage Directive
- **2014/30/EU** Electromagnetic Compatibility (EMC Directive)
- **2011/65/EU** "Restricted use of certain hazardous substances in electrical and electronic equipment" (RoHS Directive)

The EU Conformity Declarations are provided on the SIGMATEK website. See Products/Downloads or use the search function and the keyword "EU Declaration of Conformity".

4 Type Plate

Exemplary nameplate (symbol image)

HW: Hardware version
SW: Software version

5 Technical Data

5.1 Pixel LED Outputs Specifications

Number of channels	2
Short-circuit and overload protection	yes (short-circuit or overload can be read from software through a latched status bit)
Supported pixel LEDs	WS2812B, WS2813
Maximum number of pixel LEDs	512 pixel LEDs
Data rate Pixel LED control	800 kB/s
Update frequency	circa 115 Hz @ 288 pixel LEDs, 800 kB/s (the update rate is calculated from the number of pixel LEDs and the fixed update rate, if a higher update rate is required, then the number of pixel LEDs per string must be reduced)
Data width per LED	24 bit (8 bit red/green/blue)
Maximum load capacity +5 V	max. 4.5 A at 50 °C environmental temperature max. 4.0 A at 55 °C environmental temperature

5.2 24 V PWM LED Outputs Specifications

Number	2
Short-circuit and overload protection	yes (short-circuit or overload can be read from software through a latched status bit)
Maximum permissible continuous load current/channel	2.0 A
Maximum total current	4 A (100% of on-time)
PWM pulse width	0-100 % adjustable in 1 % increments
PWM frequency	adjustable 100 Hz - 2 kHz
Residual current output (off)	≤ 10 µA

5.3 Electrical Requirements

External +24 V supply	+18-30 V DC	
External current consumption +24 V	max. total current 5.4 A (2x pixel LED total current max. 1.4 A, 2x PWM output total current max. 4 A)	
Voltage supply from S-DIAS bus	+5 V	
Current consumption on the S-DIAS bus (+5 V power supply)	typically 90 mA	maximum 100 mA
Voltage supply from S-DIAS bus	+24 V	
Current consumption on the S-DIAS bus (+24 V power supply)	typically 6 mA	maximum 8 mA

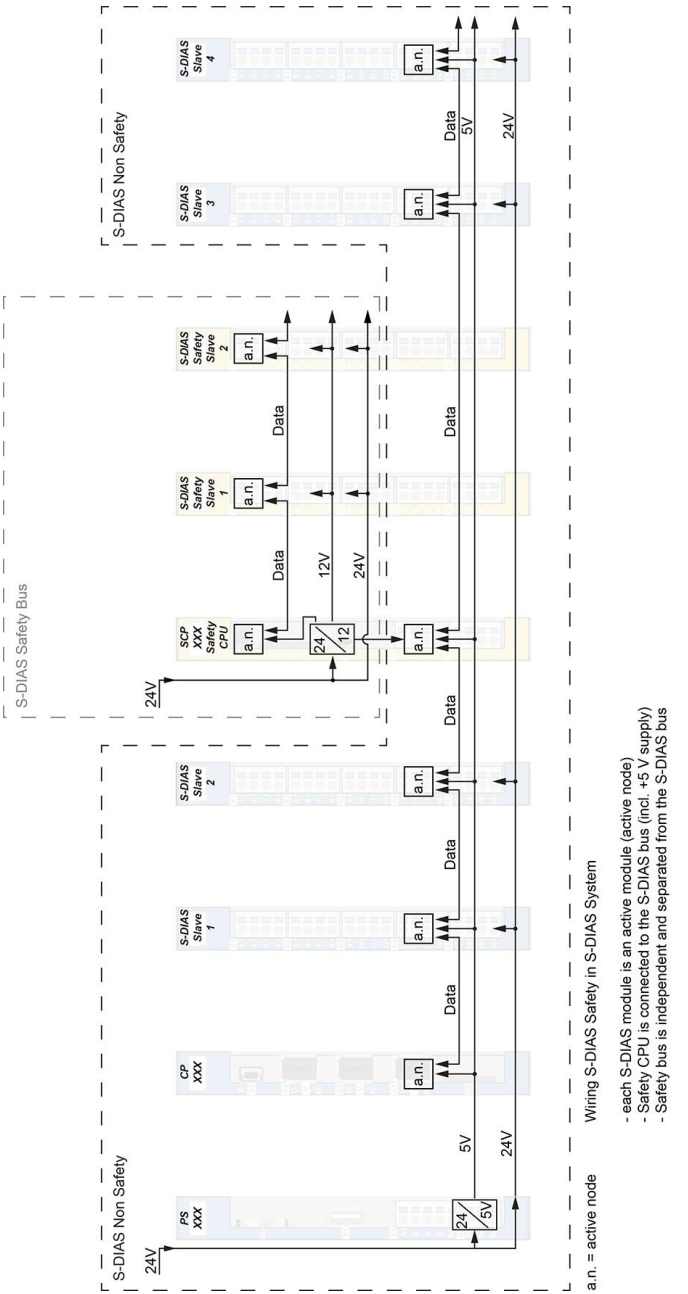
INFORMATION

If this S-DIAS module is connected to an S-DIAS supply module with several S-DIAS modules, the total current of the modules used must be determined and checked.

The total current of the +24 V supply cannot exceed 1.6 A!

The total current of the +5 V supply cannot exceed 1.6 A!

The specification for the current can be found in the module-specific documentation under "Electrical Requirements".


S-DIAS Non Safety
24V
PS XXX
24/5V
5V
24V
CP XXX
a.n.
S-DIAS Slave 1
S-DIAS Slave 2
Data
S-DIAS Safety Bus
SCP XXX Safety CPU
24/12
12V
24V
S-DIAS Safety Slave 1
S-DIAS Safety Slave 2
S-DIAS Non Safety
S-DIAS Slave 3
S-DIAS Slave 4
a.n. = active node
Wiring S-DIAS Safety in S-DIAS System
- each S-DIAS module is an active module (active node)
- Safety CPU is connected to the S-DIAS bus (incl. +5 V supply)
- Safety bus is independent and separated from the S-DIAS bus

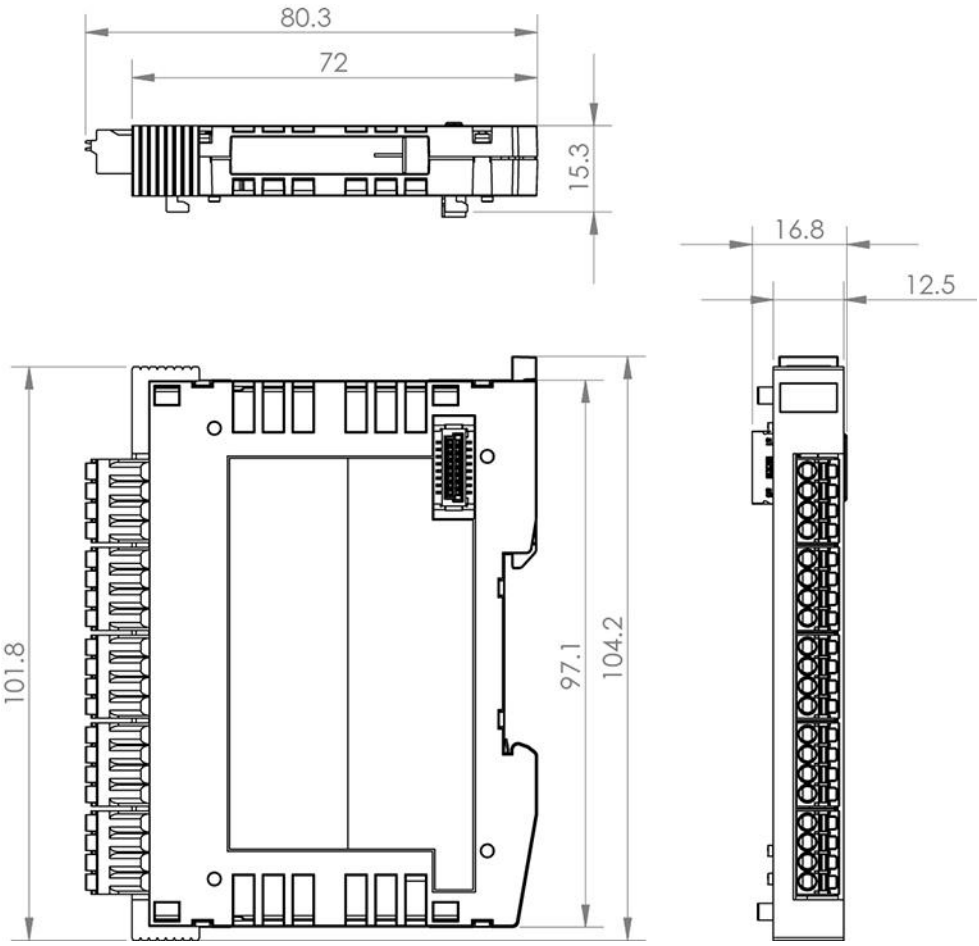
5.4 Miscellaneous

Article number	20-018-221
Approvals	CE, UKCA

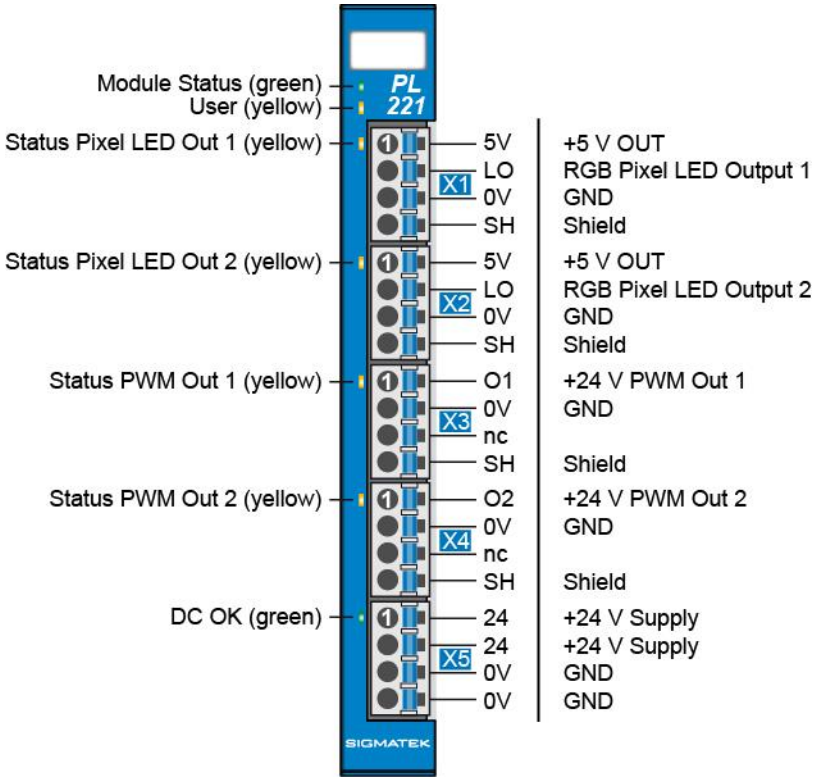
5.5 Environmental Conditions

Storage temperature	-20 ... +85 °C	
ambient temperature	0 ... +55 °C	
Humidity	0-95 %, non-condensing	
Installation altitude above sea level	0-2000 m without derating > 2000 m up to a maximum of 5000 m with derating of the maximum environmental temperature by 0.5 °C per 100 m	
Operating conditions	pollution degree 2	
EMC resistance	in accordance with EN 61000-6-2 (industrial area)	
EMC noise generation	in accordance with EN 61000-6-4 (industrial area)	
Vibration resistance	EN 60068-2-6	3.5 mm from 5-8.4 Hz 1 g from 8.4-150 Hz
Shock resistance	EN 60068-2-27	15 g
Protection type	EN 60529	IP20

6 Mechanical Dimensions



7 Connector Layout



INFORMATION

The connections of the +24 V supply (X4: pin 1 and pin 2) or the GND supply (X4: pin 3 and pin 4) are internally bridged. To supply the module, only one connection to a +24 V pin (pin 1 or pin 2) and a GND pin (pin 3 or pin 4) is required. The bridged connections may be used for further looping of the +24 V supply and the GND supply. However, it must be taken into account that a total current of 6 A per connection is not exceeded by the forward looping!

7.1 Status LEDs

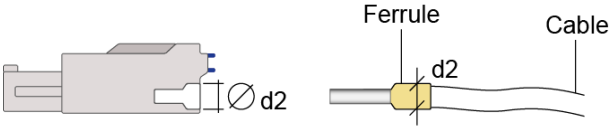
Module status	green	ON	module active
		OFF	no supply available
		BLINKING (5 Hz)	no communication
User	yellow	ON	can be set from the application (e.g. the module LED can be set to blinking through the visualization, so that it is easily found in the control cabinet)
		OFF	
		BLINKING (2 Hz)	
		BLINKING (4 Hz)	
Status pixel LED output, Status PWM out	yellow	ON	RGB pixel LED output active/+24 V PWM out active
		OFF	RGB pixel LED output inactive/+24 V PWM out inactive Short circuit or overload
DCOK	green	ON	external voltage supply OK
		OFF	no external voltage supply available

7.2 Applicable Connectors

Connectors:
X1-X5: Connectors with spring terminals (included in delivery)
The spring terminals are suitable for the connection of ultrasonically compressed (ultrasonically welded) strands.

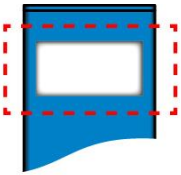
Connections:

Stripping length/sleeve length:	10 mm
Mating direction:	parallel to the conductor axis or circuit board
Conductor cross section rigid:	0.2-1.5 mm^2
Conductor cross section flexible:	0.2-1.5 mm^2
Conductor cross-section strands ultrasonically compacted:	0.2-1.5 mm^2
Conductor cross section AWG/kcmil:	24-16
Conductor cross section flexible with ferrule without plastic sleeve:	0.25-1.5 mm^2
Conductor cross section flexible with ferrule with plastic sleeve:	0.25-0.75 mm^2 (reason for reduction d2 of the ferrule)



d2 = max. 2.8 mm

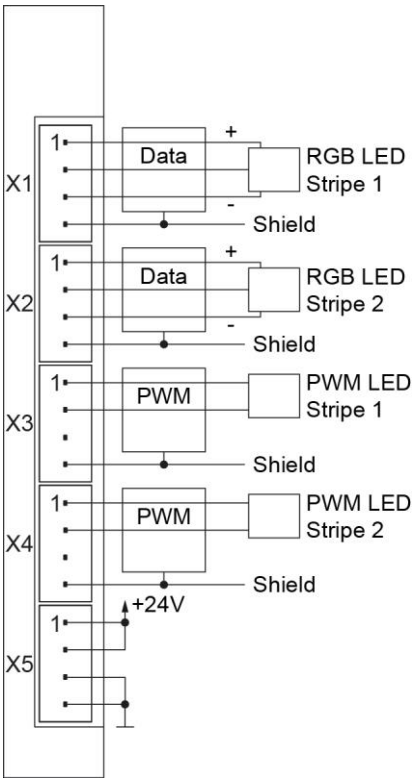
7.3 Labeling Field



Manufacturer	Weidmüller
Type	MF 10/5 CABUR MC NE WS
Article number Weidmüller	1854510000
Compatible printer	Weidmüller
Type	Printjet Advanced 230V
Article number Weidmüller	1324380000

8 Wiring

8.1 Wiring Example



8.2 Notes

To ensure error-free operation, a careful wiring method must be followed:

- The 0 V connection of the supply voltage must be connected with the 0 V collection point over the shortest route possible.
- The DIN rail must have an adequate mass connection.
- Signal lines > 3 m for the pixel RGB LEDs must be shielded.
- Signal lines > 3 m for the +24 V PWM LED control must be shielded.
- Correct wiring to ground

INFORMATION

The ground bus should be connected to the control cabinet if possible!

The S-DIAS module CANNOT be connected/disconnected while voltage is applied!

9 Assembly/Installation

9.1 Check Contents of Delivery

Ensure that the contents of the delivery are complete and intact. See chapter 1.3 Contents of Delivery.

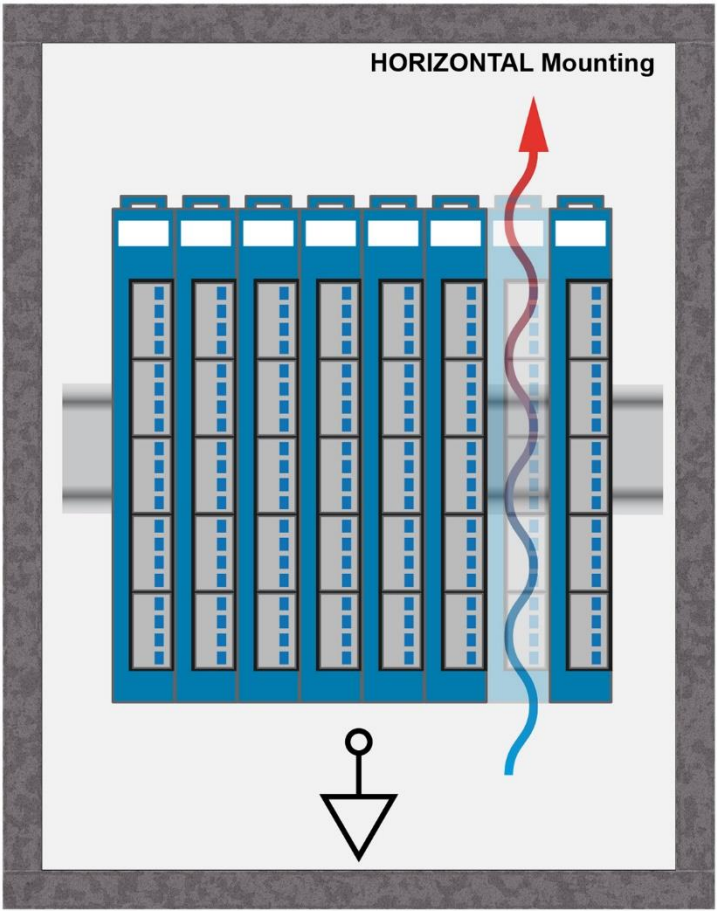
INFORMATION

On receipt and before initial use, check the device for damage. If the device is damaged, contact our customer service and do not install the device in your system.

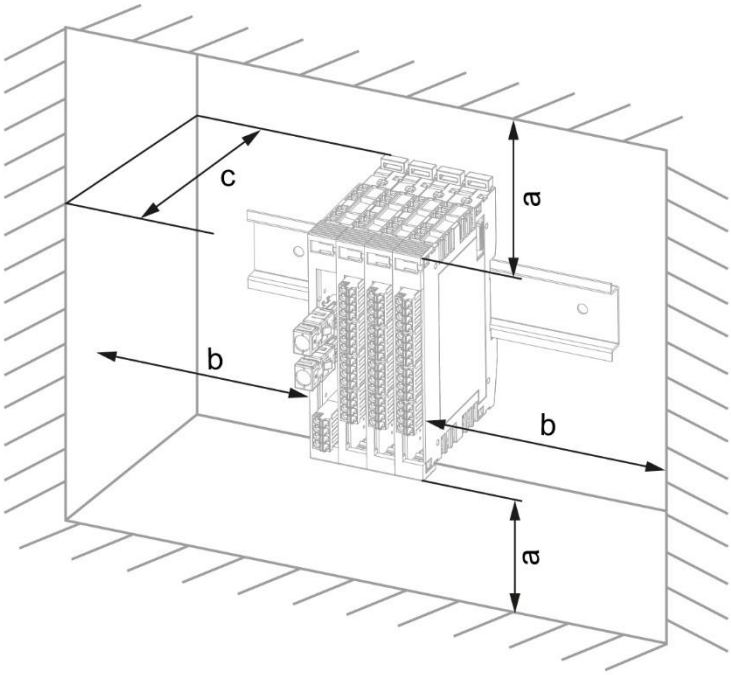
Damaged components can disrupt or damage the system.

9.2 Mounting

The S-DIAS modules are designed for installation into the control cabinet. To mount the modules a DIN-rail is required. The DIN rail must establish a conductive connection with the back wall of the control cabinet. The individual S-DIAS modules are mounted on the DIN rail as a block and secured with latches. The functional ground connection from the module to the DIN rail is made via the grounding clamp on the back of the S-DIAS modules. The modules must be mounted horizontally (module label facing up) with sufficient clearance between the ventilation slots of the S-DIAS module blocks and nearby components and/or the control cabinet wall. This is necessary for optimal cooling and air circulation, so that proper function up to the maximum operating temperature is ensured.



Recommended minimum distances of the S-DIAS modules to the surrounding components or control cabinet wall:



a	b	c
30 mm (1.18″)	**30 mm** (1.18″)	**100 mm** (3.94″)

a, b, c … distances in mm (inches)
ϑ … valid ambient temperature in °C (°F)

The specified installation distances may be reduced if appropriate measures and technical precautions are taken to dissipate the corresponding power loss.

10 Addressing

Address (hex)	Size (bytes)	Access Type	Description	Reset value
PDO				
0000	128	w	Configuration strips Bit 9..0: Pixel address (allowed: 0-681) Bit 12..10: LED strip selection “000“: LED strip 1 “001“: LED strip 2 “111“: LUT RAM Bit 15..13: Pixel mode “000“: 24-bit mode “001“: 16-bit mode “010“: 8-bit mode 011 4-bit mode	-
0002	128	w	24-bit mode (write in 3 bytes) Bit 7..0: Pixel data BLUE Bit 15..8: Pixel data GREEN Bit 23..16: Pixel data RED 16-bit mode (write in 2 bytes) Bit 4..0: Pixel data BLUE Bit 10..5: Pixel data GREEN Bit 15..11: Pixel data RED 8-bit mode Bit 7..0: Address in LUT RAM 4-bit mode Bit 3..0: Address in LUT RAM (first pixel) Bit 7..4: Address in LUT RAM (second pixel)	-

0080	4+x	w	Configuration change 1 Bit 9..0: Pixel address (allowed: 0-681) Bit 12..9: LED strip selection “000“: LED strip 1 “001“: LED strip 2 “111“: LUT RAM Bit 15..13: Pixel mode “000“: 24-bit mode “001“: 16-bit mode “010“: 8-bit mode 011 4-bit mode	-
0082	4+x	w	Pixel counter change 1 Bit 6..0: Number of pixels to change Bit 7: Another change after the pixel data	-
0083	4+x	w	24-bit mode (write in 3 bytes) Bit 7..0: Pixel data BLUE Bit 15..8: Pixel data GREEN Bit 23..16: Pixel data RED 16-bit mode (write in 2 bytes) Bit 4..0: Pixel data BLUE Bit 10..5: Pixel data GREEN Bit 15..11: Pixel data RED 8-bit mode Bit 7..0: Address in LUT RAM 4-bit mode Bit 3..0: Address in LUT RAM (first pixel) Bit 7..4: Address in LUT RAM (second pixel)	-
0xxx	4+x	w	Configuration change	-
0xxx	4+x	w	Pixel counter change x Bit 6..0: Number of pixels to change Bit 7: Another change after the pixel data	

0xxx	4+x	w	24-bit mode (write in 3 bytes) Bit 7..0: Pixel data BLUE Bit 15..8: Pixel data GREEN Bit 23..16: Pixel data RED 16-bit mode (write in 2 bytes) Bit 4..0: Pixel data BLUE Bit 10..5: Pixel data GREEN Bit 15..11: Pixel data RED 8-bit mode Bit 7..0: Address in LUT RAM 4-bit mode Bit 3..0: Address in LUT RAM (side 1) Bit 7..4: Address in LUT RAM (side 2)	-
PDO				
0100	2	r/w16	Strip 1 configuration Bit 9..0: Start address for sequencer (allowed: 0-681) Bit 14..10: reserved Bit 15: Strip 1 auto-increment	8000
0102	2	r/w16	Strip 2 configuration Bit 9..0: Start address for sequencer (allowed: 0-681) Bit 14..10: reserved Bit 15: Strip 2 auto-increment	8000
0104	1	w	Pixel output SW start Bit 0: Sequencer pixel output 1 start Bit 1: Sequencer pixel output 2 start Bit 7..2: reserved	-
105	1	w	PWM 1 output high time Bit 6..0: High time Period in % 0 = 0 % high time (PWM off) 50 = 50 % high time 100 = 100 % high time Bit 7: reserved	00

106	1	w	PWM 2 output high time Bit 6..0: High time Period in % 0 = 0 % high time (PWM off) 50 = 50 % high time 100 = 100 % high time Bit 7: reserved	00
107	1	r	Status Register Bit 0: DC 24 V OK Bit 1: DC 5 V OK Bit 2: overcurrent AUPIS (latched) Bit 3: overcurrent LEDs (latched) Bit 4: short circuit AUPIS (latched) Bit 5: over temperature AUPIS (latched) Bit 6: 5 V supply over temperature error (latched) Bit 7: reserved	00
0108	8	r/w	reserved	-
SDO				
0110	1	r/w	Pixel output configuration Bit 0: Pixel output 1 start config 0 ... SW Start 1 ... Continuous flow Bit 1: Pixel output 2 start config 0 ... SW Start 1 ... Continuous flow Bit 2: Pixel output 1 use end address Bit 3: Pixel output 2 use end address Bit 4: Pixel output 1 switch on Bit 5: Pixel output 2 switch on Bit 7..6: reserved	33
0111	1	r/w	Control register Bit 6..0: reserved Bit 7: 5 V DC switch on	00

0112	2	r/ w16	Pixel output 1 length Bit 9..0: Output length (allowed: 1-682) Bit 15..10: reserved	0001
0114	2	r/ w16	Pixel output 1 length Bit 9..0: Output length (allowed: 1-682) Bit 15..10: reserved	0001
0116	1	r/w	Pixel output configuration	0116
0117	1	r/w	reserved	
0118	1	r/w	Pixel output 1 T0H (in 20 ns) Bit 7..0: T0H high time (3-255)	03
0119	1	r/w	Pixel output 1 T1H (in 20 ns) Bit 7..0: T1H high time (3-255)	03
011A	1	r/w	Pixel output 1 T0L (in 20 ns) Bit 7..0: T0L Low time (3-255)	03
011B	1	r/w	Pixel output 1 T1L (in 20 ns) Bit 7..0: T1L Low time (3-255)	03
011C	1	r/w	Pixel output 2 T0H (in 20 ns) Bit 7..0: T0H high time (3-255)	03
011D	1	r/w	Pixel output 2 T1H (in 20 ns) Bit 7..0: T1H high time (3-255)	03
011E	1	r/w	Pixel output 2 T0L (in 20 ns) Bit 7..0: T0L Low time (3-255)	03
011F	1	r/w	Pixel output 2 T1L (in 20 ns) Bit 7..0: T1L Low time (3-255)	03
0120	2	r/ w16	Reset code time output 1 (in µs) Bit 8..0: Reset code time Bit 15..9: reserved	0032
0122	2	r/ w16	Reset code time output 2 (in µs) Bit 8..0: Reset code time Bit 15..9: reserved	0032
0124	2	r/ w16	PWM output 1 period (50-1000) * Bit 9..0: output period 50 = 2 kHz 51 = 1.96 kHz 52 = 1.92 kHz ... 998 = 100.2 Hz 999 = 100.1 Hz 1000 = 100.0 Hz Bit 15..10: reserved	0000

0126	2	r/ w16	PWM output 2 period (50-1000) * Bit 9..0: output period 50 = 2 kHz 51 = 1.96 kHz 52 = 1.92 kHz ... 998 = 100.2 Hz 999 = 100.1 Hz 1000 = 100.0 Hz Bit 15..10: reserved	0000
0128	2	r/ w16	Sequencer 1 end address Bit 9..0: End address for sequencer (allowed: 0-681) Bit 15..10: reserved	0000
012A	2	r/ w16	Sequencer 1 end address Bit 9..0: End address for sequencer (allowed: 0-681) Bit 15..10: reserved	0000

*Info: PWM frequency = 1/(output period *100 ns *100)

11 Supported Cycle Times

11.1 Cycle Times below 1 ms (in µs)

50	100	125	200	250	500
x	x	x	x	x	x

x= supported

11.2 Cycle Times equal to or higher than 1 ms (in ms)

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x

x= supported

17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32
x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x

x= supported

12 Hardware Class PL221

Hardware Class PL221 for the S-DIAS DMX module PL 221

This hardware class is used to control the PL 221 hardware module. The module has 2 pixel LED outputs and 2 PWM outputs. More information on the hardware can be found in the module documentation.

12.1 General

ClassState	State	This server shows the actual status of the hardware class.
DeviceID	State	The device ID of the hardware module is shown in this server.
FPGAVersion	State	FPGA version of the module in the format 16#XY (e.g. 16#10 = Version 1.0).
Hardware Version	State	Hardware version of the module in the format 16#XXYY (e.g. 16#0120 = Version 1.20)
Serial Number	State	The serial number of the hardware module is shown in this server.
Retry Counter	State	This server increments when a transfer fails.
LED Control	Output	With this server, the application LED of the S-DIAS module can be activated to find the module in the network more quickly. The following status are possible: 0 LED off 1 LED on 2 blinks slowly 3 blinks rapidly
Required	Property	This client is active by default, which means that the S-DIAS hardware module at this position is mandatory for the system and can under no circumstances be disconnected or return an error. Otherwise, the entire hardware deactivated. If the hardware module is missing or removed, an S-DIAS error is triggered. If his client is initialized with 0, the hardware module located in this position is not mandatory. This means that it can be inserted or removed at any time. However, which components identified as "not required" should be selected with regard to the safety of the system.
Range	State	This server shows whether an overcurrent, short circuit or overtemperature has occurred at the LED output or PWM output. Bit 0 overcurrent PWM Bit 1 overcurrent LEDs Bit 2 short circuit PWM Bit 3 overtemperature PWM Bit 4 5 V supply overtemperature error The status is queried over read(). During the status query, the server is reset to 0, if the error is no longer active.
ExternVoltage-OK	State	This server indicates whether the external module supply for 24 V and 5 V is OK. Bit 0 24 V OK Bit 1 5 V OK

12.2 LED Pixel Output

LEDStripeOn	Output	This server switches the LED stripes on/off. 0 LEDs are deactivated 1 LEDs are activated

SetAllLed	Output	With this server the entire LED stripe can be set to one color. 0 normal use 1 setting stripe 1 2 setting stripe 2 3 setting both stripes
SetColorStripe[1-2]	Output	This server is used to set the color for the “SetAllLed” mode. The entire LED stripe is set to the defined color. E.g.: 0xFFFFFF (24 bit); 0xFFFF (16 bit); 0-255 (8 bit); 0-15 (4 bit)
StartRollingAddrStripe[1-2]	Output	This server is used to start the chaser. (see Internal Properties) -1 off 0-(number of LEDs -1) valid start address
Stripe[1-2]_Activate	Property	On this client, the LED stripe can be deactivated/activated. As initialization value 0 stripe [1,2] is deactivated 1 stripe [1,2] is activated as initialization value
Stripe[1-2]_NumOfLeds	Property	This client is used to configure the LED stripes. How many LEDs are on the stripes used. Maximum: 512 as initialization value
Stripe[1-2]_Type	Property	This client is used to configure the LED stripe. Which strip type is used. 0 WS2812B 1 WS2813 2 Individual For individual configuration, the required parameters must be entered using the IndividualStripeConfig function. (see Internal Properties) as initialization value
Stripe[1-2]_ColorResolution	Property	This client can be used to select the color resolution for transmission. 0 24-bit 1 16-bit 2 8-bit (LUT) 3 4-bit (LUT) In the 8 bit or 4 bit setting, the colors of the default color table or the user color table connected to the client "ColorTable" are used. This enables data reduction during transmission. (see Internal Properties) as initialization value
Stripe[1-2]_RefreshCycle	Property	Specifies the frequency at which the LED stripes are updated when WriteDataMode = 0 is selected. In order to maintain an even workload, the data is split up. The time must be a multiple of the bus cycle time. E.g.: 50 Hz ≙ 20 ms as initialization value
ColorTable	Property	This client can be connected to an object of the PL221ColorTab class (see class description PL221ColorTab). If the default color table is used, no PL221ColorTab object must be connected. as initialization value if necessary

WriteDataMode	Property	This client is used to select how the data transfer should take place. In mode 0 a complete memory for a LED stripe can be transferred. The data are sent to the module in the defined refresh time. With mode 1, individual LED changes can be made for both LED stripes. e. g.: change 4 LEDs to offset 7 + 2 LEDs to offset 24. These are collected and transferred per cycle to a maximum of 128 bytes. It must be taken into account that the strip configuration of 3 bytes is also sent for each change. (more information, see Internal Properties) 0 entire LED stripes 1 single LED changes as initialization value

12.3 PWM Output

PWMOut [1-2]	Output	Defines the PWM switch-on time for channel 1-2. Switch-on time is given as a percentage from 0 to 100 %. 0 PWM off 100 100 % on
PWMEnable[1-2]	Property	The PWM outputs can be activated/deactivated in this client. As initialization value Possible values: 0 PWM output is deactivated 1 PWM output is activated
PWMPeriode[1-2]	Property	Defines the period duration of the PWM output. (50-1000) in 10 µs steps. Possible values: 50 2 kHz 51 1,96 kHz 52 1,92 kHz ... 998 100,2 Hz 999 100,1 Hz 1000 100,0 Hz as initialization value

12.4 Communication Interfaces

ALARM	Downlink	With this downlink the corresponding alarm class can be placed via the hardware editor.

12.5 Global Methods

The following methods can be called via the ClassState server.

12.5.1 ChangeIndividualLED

This method can only be used if the client **WriteDataMode** has been set to **1**.

Transfer parameters	Type	Description
usStripeNumber	USINT	Indicates the LED strip that is being written with the data.
uiPixelStartAddr	UINT	Each LED is assigned to an address in the RAM. This parameter is used to specify the LED from which the change is to be saved.
usNumberOfLed	USINT	Defines how many LEDs should be changed.
pData	^void	A pointer to an array with the desired pixel color data must be passed.
dStartOffset	DINT	With this parameter the stored color pattern can be moved to another position of the LED strip. The color pattern in the RAM is not affected by this. dStartOffset cannot be greater than the specified LED number.
Return parameters	**Type**	**Description**
dRetCode	DINT	0 data have been copied -1 entry is larger than 125 bytes (+ 3 bytes header) -2 entry is too large for remaining DO size -3 wrong WriteData mode selected -4 dStartOffset is greater than the number of used LEDs -5 ClassState not OK -6 invalid pointer to the color data -8 invalid stripe selection

This function is used to change individual LEDs on both stripes. Only changes of up to 125 bytes are possible. (for more information, see Internal Properties)

12.5.2 ChangeAllLEDs

This method can only be used if the client **WriteDataMode** has been set to **0**.

Transfer parameters	Type	Description
usStripeNumber	USINT	Indicates the LED strip that is being written with the data.
uiPixelStartAddr	UINT	Each LED is assigned to an address in the RAM. This parameter is used to specify the LED from which the change is to be saved.
uiNumberOfLed	UINT	Specifies how many LEDs are to be written.
pLedData	^void	Pointer to the array, where the pixel data are stored.
dStartOffset	DINT	With this parameter the stored color pattern can be moved to another position of the LED strip. The color pattern in the RAM is not affected by this. dStartOffset cannot be greater than the specified LED number.
Return parameters	**Type**	**Description**
dRetCode	DINT	0 data have been copied -1 wrong WriteData mode selected -4 dStartOffset is greater than the number of used LEDs -5 ClassState not OK -6 invalid pointer to the color data -8 invalid stripe selection

This function is used to write a whole stripe RAM. The LED data is split and copied in the given time to achieve an even load. (for more information, see Internal Properties)

12.5.3 SetWholeStripe

This method can be used independently of the set **WriteDataMode**.

Transfer parameters	Type	Description
usSelectStripe	USINT	Specifies which LED stripe should be set to the desired color. 1 stripe 1 2 stripe 2
pColorData	^void	Pointer to the array, where the pixel data are stored.
bEnable	BOOL	Activates/Deactivates the function. 1 activate (default) 0 deactivate Deactivate is used to reset the LED strip to the stored color pattern in the RAM.
Return parameters	**Type**	**Description**
dRetCode	DINT	WriteDataMode = 1: 0 data have been copied -1 entry is larger than 125 bytes (+ 3 bytes header) -2 entry is too large for remaining DO size -6 invalid pointer to the color data WriteDataMode = 0: 0 data have been copied -5 ClassState not OK -6 invalid pointer to the color data -8 invalid stripe selection

This function is used to set one or both LED strips to a uniform color. The color pattern previously stored in the RAM is not affected by this function. If the function is deactivated again with bEnable = 0, the color pattern previously stored in the RAM is displayed again.

12.6 Private Methods

12.6.1 IndividualStripeConfig

If the value 2 is set at the client Stripe_Type[1,2], the function IndividualStripeConfig must be overloaded and the correct values of the used stripe type must be entered.

Needed values:

- Pixel order: [RGB; RBG; GRB; GBR; BGR; BRG]
- LSBFirstOut: Which data byte is output first.
  - 0 .. Bit 7
  - 1 .. Bit 0
- PixelOutT0H, PixelOutT1H, PixelOutT0L, PixelOutT1L: Data transfer time [µs]
- ResetPixelTime: Data transfer reset time [µs]

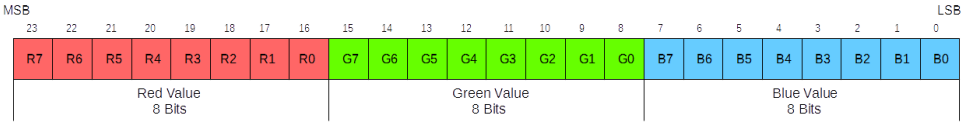
12.7 Internal Properties

12.7.1 Setting the Color Resolution

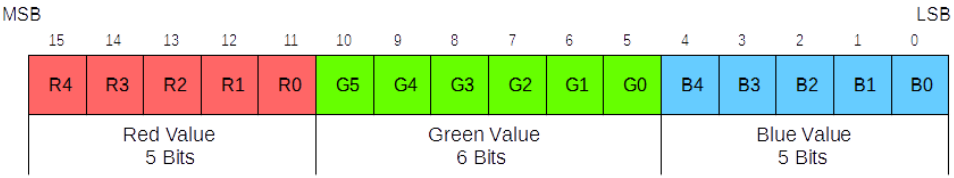
You can select 4 different data transfer settings. So this allows to specify, how many bytes are used to describe a color. In the 24 bit setting, 3 bytes of color data are transferred per LED, so that 2^{24} colors are possible for display. In the 16 bit setting, 2 bytes of color data are transferred per LED, so that 2^{16} colors are possible for display. In the 8 bit or 4 bit setting, the color data is obtained from a 256 byte default or user color table that has been previously defined. In the 8 bit setting, only 1 byte per LED is required and can display all 256 different predefined colors stored in the color table. In the 4 bit setting, two colors are transmitted in one byte and thus 16 (2^4) can be displayed. Here, the first 16 colors of the color table are used.

With the setting of 8 bit and 4 bit, the colors in the table are addressed by decimal numbers. The specified number corresponds to the memory location in the table.
Example with the default table:
8 bit [0-255]: Input 9 -> color red
4 bit [0-15]: Input 15 -> color white

Memory structure:
24 bit:



16 bit:



Example with 6 LEDs:
24 bit: 6 LEDs * 3 bytes = 18 bytes of data
16 bit: 6 LEDs * 2 bytes = 12 bytes of data
8 bit: 6 LEDs * 1 byte = 6 bytes of data
4 bit: (6 LEDs * 1 byte) / 2 = 3 bytes of data

12.7.2 Explanation of the methods ChangeAllLEDs and ChangeIndividualLED

The LED data is written to a separate RAM for each LED stripe. The RAM is selected with the transfer parameter usStripeNumber. The parameter uiPixelStartAddr specifies where in the RAM the color data are written and thus which LED is written with this color. With the ChangeAllLEDs function, the data packets are evenly distributed in order to achieve a uniform bus load. The client Stripe_RefreshCycle can be used to specify the time interval at which the packets are to be sent. With the ChangeIndividualLED function, individual LED changes can be made to both strips. A maximum of 128 bytes can be transferred per bus cycle. For each function call, 3 bytes of header are sent for the strip configuration!

The module should only be controlled via the global methods. The servers serve only for minimal control of the module. Inconsistent data can occur if methods and servers are used simultaneously.

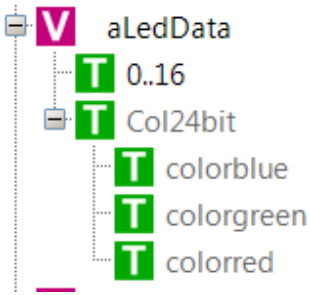
ChangeIndividualLED():
Example for 24 bit setting:
maximum LED changes with one function call per cycle:
41 LEDs * 3 bytes + 3 bytes header = 126 bytes

maximum LED changes with two function calls per cycle:
(20 LEDs * 3 bytes + 3 bytes header) + (20 LEDs * 3 bytes + 3 bytes header) = 126 bytes

Code example for a function call:
At the first LED strip, from the 5th LED, 10 LEDs are changed. The pattern is not moved because the sStartOffset is 0.

```
retcode := toPL221.ChangeIndividualLED(usStripeNumber   :=1
                                     , uiPixelStartAddr:=5
                                     , usNumberOfLed    :=10
                                     , pData            :=#aLedData[0]
                                     , dStartOffset     :=0);
```

Example for the transfer array of the LED data:
24 bit setting:
An array for 17 LEDs is filled with the color white. This array is then transferred in the function ChangeAllLEDs or ChangeIndividualLED.



Properties

Name	aLedData
GUID	NULL
Type	ARRAY OF Col24bit
Element Type	Col24bit
Pointer	false
Comment	

```
for i:= 0 to 16 do
  aLedData[i].colorred    := 255;
  aLedData[i].colorgreen  := 255;
  aLedData[i].colorblue   := 255;
end_for;
```

12.7.2.1 16 bit setting

With this setting, only 2 bytes of color data are transferred per LED (see memory structure 16 bits). The array (ARRAY OF UINT) is filled with the color white for 17 LEDs and then transferred in the function ChangeAllLEDs or ChangeIndividualLED.

```
for i:= 0 to 16 do
  aLedData16b[i] := (255 AND 0x1F)              //blue
                 OR ((255 SHL 5) AND 0x7E0)     //green
                 OR ((255 SHL 11) AND 0xF800); //red
end_for;
```

12.7.2.2 8 bit setting

An array (ARRAY OF USINT) with 17 LEDs is filled with the first 17 color values of the default LookUp table. This array is then transferred in the function ChangeAllLEDs or ChangeIndividualLED. The server LUTAddress specifies the start address of the color table, in this example 0.

```
for j:= 0 to 16 do
  aLedData8bit[j]  := LUTAddress$USINT;
  LUTAddress += 1;
end_for;
```

12.7.2.3 4 bit setting

In this setting, 2 color data are transferred per byte. The first color address for the first LED in the lower 4 bits and the second color address in the upper 4 bits for the second LED. This array is then transferred in the function ChangeAllLEDs or ChangeIndividualLED.

```
aLedData8bit[0] := (0 AND 0x0F) OR ((1 SHL 4) AND 0xF0);
```

12.7.3 Use of the chaser with the Parameter dStartOffset or the Server StartRollingAddrStripe1,2

12.7.3.1 Parameter dStartOffset

The LED pattern previously stored in RAM can be offset using the ChangeIndividualLED or ChangeAllLEDs functions. If the function is called cyclically, a running light is generated. The direction of the running light is determined by the offset. Depending on whether the offset is increased or decreased.

For the direction **right to left** the offset is increased up to the maximum LED number -1. The maximum number is equal to the client value StripeX_NumOfLeds.

```
Retcode := toPL221.ChangeIndividualLED(usStripeNumber:=1
                        , uiPixelStartAddr:=PixelStartAddr$UINT
                        , usNumberOfLed:=NumOfLed$USINT
                        , pData:= #aLedData24Bit[0]
                        , dStartOffset:=StartOffset
                        );

//Running Ligth from Right to Left
//MaxNumbersOfLed is value from Client Stripe1_NumOfLeds
if StartOffset < (MaxNumberOfLeds-1) then
  StartOffset += 1; //increment offset
else
  StartOffset := 0;
end_if;
```

For the direction left to right the offset is reduced, with start value = maximum LED number -1. The maximum number is equal to the client value StripeX_NumOfLeds.

```
Retcode := toPL221.ChangeIndividualLED(usStripeNumber:=1
                                     , uiPixelStartAddr:=PixelStartAddr$UINT
                                     , usNumberOfLed:=NumOfLed$USINT
                                     , pData:= #aLedData24Bit[0]
                                     , dStartOffset:=StartOffset
                                     );

//Running Ligth from Left to Rigth
//MaxNumbersOfLed is value from Client Stripe1_NumOfLeds
if StartOffset > 0 then
  Startoffset -= 1; //decrement offset
else
  Startoffset := MaxNumberOfLeds-1;
end_if;
```

12.7.3.2 Server StartRollingAddrStripe1,2

The LED pattern previously stored in RAM can be started as a running light by cyclically calling the Write method and incrementing the server. The pattern only needs to be written into the RAM stripe once. The speed of the chaser is determined by the frequency of calls.

Example:
An LED stripe with 10 LEDs is written using the ChangeIndividualLED function. The first 4 LEDs white, 2 LEDs red and again 4 LEDs white. If you increment the server every cycle, you can see how the red LEDs move along the whole stripe. This function does NOT work if the stripe is set to one color with the server "SetAllLeds".

Code example for a cyclic call of a running light:

```
FUNCTION VIRTUAL GLOBAL PL221Test::CyWork
  VAR_INPUT
    EAX   : UDINT;
  END_VAR
  VAR_OUTPUT
    state (EAX)   : UDINT;(* := READY *)
  END_VAR

  //cyclictime = 50 ms

  //wait for start the Rolling mode
  if StartRunninglight = 1 then
      //calling the Write Methode
      toPL221.StartRollingAddrStripe1.Write(input:= RunningCounter MOD Stripe1_NumOfLeds);
      //increment counter
      RunningCounter += 1;
  else
    //stop Rolling mode
    toPL221.StartRollingAddrStripe1.Write(input:= -1);
    RunningCounter := 0;
  end_if;

END_FUNCTION
```

13 Transport/Storage

INFORMATION

This device contains sensitive electronics. During transport and storage, high mechanical stress must therefore be avoided.

For storage and transport, the same values for humidity and vibration as for operation must be maintained!

Temperature and humidity fluctuations may occur during transport. Ensure that no moisture condenses in or on the device, by allowing the device to acclimate to the room temperature while turned off.

When sent, the device should be transported in the original packaging if possible. Otherwise, packaging should be selected that sufficiently protects the product from external mechanical influences. Such as cardboard filled with air cushioning.

14 Storage

INFORMATION

When not in use, store the operating panel according to the storage conditions. See chapter 13.

During storage, ensure that all protective covers (if available) are placed correctly, so that no contamination, foreign bodies or fluids enter the device.

15 Maintenance

INFORMATION

During maintenance as well as servicing, observe the safety instructions from chapter 2 Basic Safety Directives.

15.1 Service

This product was constructed for low-maintenance operation.

15.2 Repair

INFORMATION

In the event of a defect/repair, send the device with a detailed error description to the address listed at the beginning of this document.

For transport conditions, see chapter 13 Transport/Storage.

16 Disposal

INFORMATION

Should you need to dispose of the device, the national regulations for disposal must be followed.

The device appliance must not be disposed of as household waste.

Documentation Changes

Change date	Affected page(s)	Chapter	Note
16.05.2018	6	1.4 Miscellaneous	Standard and Approvals adjusted
20.09.2018		3 Connector Layout	Note added
14.11.2019	22	7 Supported Cycle Times	Chapter added
28.02.2020	22	7 Supported Cycle Times	Text adapted
08.09.2020	24	8 Hardware Class PL221	Chapter added
04.11.2020	15	5 Mounting	Expansion functional ground connection
06.12.2022	7	1.4 Miscellaneous	UKCA conformity
26.07.2023		Document	General chapters added, design