

flexyPage displays

# User Manual

Commissioning and setup

Version 2.1.x



# Content

## [Changes to version 1.39.x](#)

[New functions](#)

[Changes](#)

[Bugfixes](#)

[Known bugs](#)

## [A brief overview](#)

## [Video tutorials](#)

## [Indications on the flexyPage documentation](#)

## [Safety guidelines](#)

[Safety guidelines and restrictions](#)

## [Commissioning](#)

[Commissioning of the hardware](#)

[Required wiring](#)

[Standard configuration of inputs and outputs for external I/O modules](#)

[Commissioning of the software](#)

[Power-up](#)

[Touchscreen calibration](#)

[First login](#)

[User roles](#)

[Setup](#)

[Layout](#)

[Content](#)

[Update](#)

[Setup](#)

[The configuration interface](#)

## [First Steps](#)

[Creating a configuration in three steps](#)

[Layout selection](#)

[Background image](#)

[Color scheme](#)

[Creating your own configuration](#)

[Configuration of the display settings](#)

[Configuration of the building](#)

[Adjusting widget properties](#)

## [Device overview and connection information](#)

[Connection status](#)

## [Maintenance](#)

## [Firmware updates](#)

[Online update](#)

[USB-Update](#)

[Upload a firmware file](#)

[Update dialogue](#)

## [Calibration of the touchscreen](#)

## [Hardware settings](#)

### [Localization and time](#)

[Time](#)

[Area/Time zone](#)

[Language](#)

### [Screen](#)

### [Network settings](#)

[Network settings of the flexyPage display](#)

[LAN / IPv4 settings in the flexyPage display](#)

[Factory settings](#)

[Reserved IP addresses](#)

[Changing the IPv4 address](#)

[Displaying the network settings in an emergency](#)

### [Cellular \(4G\)](#)

[Mobilcom-Debitel \(Vodafone Netz\)](#)

[LIDL Connect](#)

[Telekom M2M](#)

### [Network services](#)

[NTP Settings](#)

[Proxy server settings](#)

## [Remote diagnosis](#)

## [Export/Import](#)

## [General display settings](#)

[Distance between widgets](#)

[Display rotation](#)

[Grid units](#)

[System information at start of programme](#)

## [Special Text](#)

### [General](#)

[Deactivation of special signals or functions](#)

[Priorities](#)

## [CAN Configuration](#)

[Lift configuration](#)

[Simulation](#)

[Baud rate](#)

[Nodes-ID](#)

[Floor](#)

[Assigned lifts](#)

[Door](#)

[Load sensor settings](#)

[Position settings](#)

[In-/Outputs](#)

[ELFIN Call Controller](#)

[CANopen master](#)

[Node-ID](#)

[General I/O parameters](#)

[External CANopen inputs and outputs](#)

[Generate messages from position values](#)

[Example topology and node-ID assignment](#)

[Energy saving functions](#)

[LVDS](#)

[CANopen-Lift](#)

[Discrete signals / inputs](#)

[Sound settings](#)

[Widgets](#)

[General Information](#)

[All buttons and switches](#)

[Widget-Descriptions](#)

[Lift-Buttons](#)

[PIN Pad](#)

[Car Calls](#)

[CarCall\[v2\]](#)

[CarCall\[h2\]](#)

[CarCall\[floor\]](#)

[CarCall\[tenant\]](#)

[CarCall\[basic\]](#)

[CarCall\[Image\]](#)

[The CarCall\[Image\] widget provides a call button that loads its background images from an URL.](#)

[Hall Calls](#)

[HallCall\[basic\]](#)

[Guest calls](#)

[GuestCall\[basic\]](#)

[Destination calls](#)

- [DestinationCallButton\[floor\]](#)
- [Buttons + Switches](#)
  - [SpecialFunctionButton\[basic\]](#)
- [KeyPad for floor selection](#)
  - [KeyPad\[character\]](#)
  - [KeyPad\[enter\]](#)
  - [KeyPad\[clear\]](#)
  - [KeyPad\[display\]](#)
- [Lift-Status Indicator](#)
  - [Direction Indicators](#)
    - [DirectionIndicator\[double\]](#)
    - [DirectionIndicator\[single\]](#)
    - [EvacuationMovingDirection](#)
    - [DirectionIndicator\[rotation\]](#)
  - [Destination call indicators](#)
    - [AcknowledgedFloors\[horizontal list\]](#)
    - [AssignedLift\[text\]](#)
  - [Announcements](#)
    - [LiftInfo\[icon+text\]](#)
    - [LiftInfo\[text\]](#)
    - [LiftInfo\[icon\]](#)
    - [SimpleMessageIndicator\[text\]](#)
  - [Floor Indicators](#)
    - [FloorIndicator\[alias\]](#)
    - [FloorIndicator\[bar\]](#)
    - [FloorIndicator\[nostalgic\]](#)
    - [SimpleFloorIndicator\[text\]](#)
  - [Lift-Status](#)
    - [LiftStatus\[classic\]](#)
  - [Info + Adverts](#)
    - [FloorInfo\[v2\]](#)
    - [FloorInfo\[stream\]](#)
- [Entertainment](#)
  - [Campaign\[basic\]](#)
- [Web services](#)
  - [Messenger\[basic\]](#)
  - [Tenants\[basic\]](#)
  - [RssReader\[header\]](#)
  - [Weather\[icon\]](#)
  - [Weather\[text\]](#)
- [Multimedia](#)
  - [TextArea\[basic\]](#)

[DiaShow\[basic\]](#)  
[VideoPlayer\[basic\]](#)  
[RemoteViewer\[image\]](#)  
[Video\[Stream\]](#)  
[FloorInfo\[video\]](#)

## Sensors

### Load

[Load\[digit\]](#)  
[Load\[needle\]](#)  
[Load\[icons\]](#)  
[Load\[bar\]](#)

### Speed

[Speed\[digit\]](#)

### Position

[Position\[shaft\]](#)  
[Position\[digit\]](#)  
[DistanceToFloor\[digit\]](#)  
[DistanceToAdjacentFloors\[compact\]](#)  
[PositionImage\[animated\]](#)

### Energy consumption

[EnergyConsumed\[digit\]](#)  
[EnergyFedBack\[digit\]](#)  
[PowerSum\[digit\]](#)  
[PowerSum\[bar\]](#)  
[PowerSum\[needle\]](#)

## Date + Time

[Date+Time\[text\]](#)  
[Date+Time\[stored\]](#)  
[Time\[analogue\]](#)  
[Sample application for the widget time \[analogue\]](#)

## Miscellaneous

[TextArea\[locked\]](#)  
[NextStopIndex](#)  
[JourneyCounter\[digit\]](#)  
[JourneyDurationCounter\[digit\]](#)

## Games

[Tic Tac Toe](#)

## Quick start guides

### Configuration of IOs

[CAN configuration set up](#)  
[IO mappings set up](#)

[Show signal](#)

[Common errors and solutions](#)

[Lift & CAN](#)

[Display](#)

[Glossary](#)

[Widget](#)

[Your contact persons](#)

# Changes to version 1.39.x

## New functions

- Widget "CarCall[Image]"
- Widget "XmlViewer[filter]"
- Option to configure your own hostnames for the displays (zeroconf)T

## Changes

- Option to upload your own certificates
- Support for special functions for destination calls in the keypad widgets
- Support for external buttons and signalers for the KeyPad widget
- Advanced text formatting options
- Possibility to visualize the status of the door in the "Position[shaft]" widget
- Various adjustments and improvements

## Bugfixes

- Fixed a bug when displaying widget context menus
- Fixed an error when displaying the PinPad in landscape view
- Fixed an error with the automatic image selection of newly added files
- Various adjustments and improvements

## Known bugs

- the *size arrows* parameter is not displayed correctly in the flexyPage Editor when a SVG file is used

## A brief overview

flexyPage is a modern, flexible system for the simultaneous display of lift information and multimedia presentations inside and outside elevators.

The flexyPage displays are available in various sizes (from 7" to 65") and with the mediaplayer in the boxed version, standard monitors and TVs, can also be controlled via a HDMI interface. Via the integrated CAN interface external sensors, input / output modules and controllers, which support the

standard CANopen CiA 417 - CANopen-Lift, can be connected. The audio signal is available via a line-out interface or a loudspeaker connection. For the input of commands by the user, different touch sensors can be connected. The configuration and connection to the internet takes place via the LAN interface or an internal 4G modem.

The displays can be used in all elevators for a wide range of applications, both in the new building and in retrofitting.

The functions and the layout of the flexyPage displays are freely configurable via the web editor described in this user manual and also via the internet. A configuration software is not necessary for this. Any current Internet browser can be used.

## Video tutorials

You can get a quick start with our video tutorials on the following topics:

- Connect to the display (Windows):  
<https://flexypage.de/en/how-connect-display-ethernet-your-windows-personal-computer>
- Connect to the display (Mac):  
<https://flexypage.de/en/how-connect-display-ethernet-your-mac>
- Log in to the user interface:  
<https://flexypage.de/en/how-login-your-display>
- Import and export the display configuration:  
<https://flexypage.de/en/how-ex-and-import-screen-designs-bin-files>
- Change the network settings:  
<https://flexypage.de/en/how-change-your-displays-network-settings>
- Change and adapt the tenants:  
<https://flexypage.de/en/how-set-or-change-your-tenants>
- Adjusting images:  
<https://flexypage.de/en/how-change-images>
- Customize the elevator information:  
<https://flexypage.de/en/how-change-your-elevators-information>
- Finding the serial number:  
<https://flexypage.de/en/where-find-displays-serial-number-and-update-status>

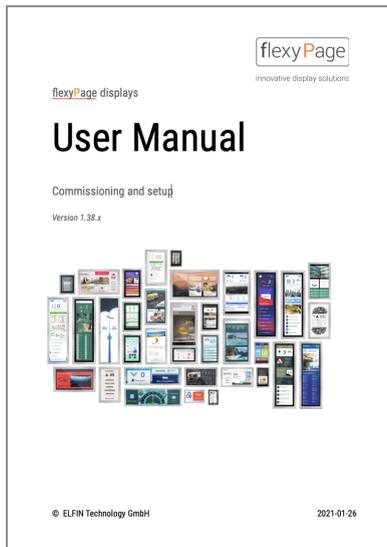
More videos and instructions can be found at <https://flexypage.de/en/videos-and-tutorials>



Do you have any questions or suggestions? Please don't hesitate to contact us at [sales@flexyPage.de](mailto:sales@flexyPage.de).



# Indications on the flexyPage documentation



This user manual describes how to setup and configure the flexyPage displays. It is only a part of the entire product documentation and subject to a constant update.

This document, the photographs, graphics and videos as well as hardware and software, are copyrighted. They may not be copied or circulated without written permission. Translations are also subject to written approval.

The contact person for copies, translations and all similar requests is ELFIN Technology GmbH.

The documentation is prepared with great care. Nevertheless, mistakes cannot be excluded. We are excited about suggestions and hints for errors in our documentation. Please contact our sales department or support.

ELFIN Technology GmbH is not liable for damages, possible errors and their consequences that arise in connection with the delivery or use of this product data sheet.

Please read the documents carefully before use!



The current version of the manuals, product data sheets as well as further documents and application examples can be found on the product homepage at:

Introduction to the flexyPage system	<a href="https://flexypage.de/en/documentation">flexypage.de/en/documentation</a>
Quick start guide	<a href="https://flexypage.de/en/documentation">flexypage.de/en/documentation</a>
Video instructions	<a href="https://flexypage.de/en/videos-and-tutorials">flexypage.de/en/videos-and-tutorials</a>
flexypage user manual	<a href="https://flexypage.de/en/user-documentation">flexypage.de/en/user-documentation</a>
Widget descriptions	<a href="https://flexypage.de/en/doc/widget-descriptions">flexypage.de/en/doc/widget-descriptions</a>
Product data sheets	<a href="https://flexypage.de/en/documentation">flexypage.de/en/documentation</a>
Frequently Asked Questions	<a href="https://flexypage.de/en/faq">flexypage.de/en/faq</a>
Contact for sales	<a href="https://flexypage.de/en/sales">flexypage.de/en/sales</a>
Contact to support	<a href="https://flexypage.de/en/support">flexypage.de/en/support</a>

# Safety guidelines

## Safety guidelines and restrictions



Please follow all the instructions in this manual and on the device.

An exclamation mark in a triangle indicates special indications on the device.

The lightning flash with an arrow indicates dangerous voltage. Ignoring this sign can result in danger to life. Installation and commissioning may only be done by professional staff! Repair of displays may only be done by ELFIN support. Autonomous opening of the casing can lead to damage on the display and an immediate loss of warranty claims. In case of damage at delivery do not plug it in! Please contact the ELFIN support.

Please, do not use harsh, abrasive cleaning agents when cleaning the device or sharp objects against the glass. Heat accumulation can lead to an overheating of the flexyPage display. This can lead to damage on the display. The cooling of the internal electronics happens passively via the casing. Provide sufficient air circulation around the flexyPage display. If the casing has ventilation slots, always make sure they are open. Wetness or fluids getting into the flexyPage display can cause electric shocks or short circuits. Only use the displays inside of buildings. Don't let fluids get inside the display.

# Commissioning

## Commissioning of the hardware

The mounting and electrical installation can differ according to the design and variant of the displays. You will find the corresponding instructions in the product data sheets of the flexyPage displays.

### Required wiring



#### **Warning!**

The wiring of the displays may only be carried out by specialists!  
Incorrect wiring can lead to injury, fire or damage to the components.

For the operation of the displays, the following wiring must be carried out before switching on for the first time:

- 24 VDC power supply (For devices of the type fel1C and fel2B 12 VDC)
- CAN bus (note termination!)
  - to the CAN bus of a lift system with a CANopen-Lift
  - or to external I/O modules + sensors for providing the signals
- RS485 interface for connection to systems with proprietary protocol
- with parallel wiring inputs/outputs and calls
- HDMI (only boxed versions)
- external speakers or active speakers
- LAN network
  - for connection to the internet or an intranet
  - for the configuration of the display
- WLAN USB adapter (fel2, fel4 and fel5)
  - Optional for activation of the access point
  - For display configuration

#### **Note LAN:**

Previously, it was necessary to use a crossover LAN cable for a point-to-point connection between two computers. For some time, the network interfaces of the devices are independently able to assign the

wires of the cable in order to establish communication. This eliminates the need for special crossover cables.

#### **Notes WLAN:**

- When a USB/WLAN adapter is detected, the supporting devices automatically set up a WLAN access point. This is the case when the adapter is plugged into the USB socket or the flexyPage device is restarted with the adapter plugged in.
- After one hour, the WLAN access point is automatically switched off again for security reasons.
- The name of the network (SSID) is the serial number, for example "fel5F-00033". This allows multiple flexyPage devices in an installation to be equipped with a wireless adapter.
- After connecting to the WLAN network, the flexyPage device can be accessed at the IP address 192.168.66.

Unfortunately, not all adapters available on the market support the access point mode or are compatible with the flexyPage devices. The following chips were successfully tested:

- Atheros AR9271 (e.g. Netgear WNA1100, TP-Link TL-WN722N, Unex DNUA-93F)
- MediaTek / Ralink RT5370

It should be noted that even adapters with the same brand and model may use different (incompatible) hardware internally without this being apparent from the outside. Accordingly, you should pay attention to the chipset when buying.

### **Standard configuration of inputs and outputs for external I/O modules**

If the flexyPage displays are used in older elevators that do not yet support the standard CANopen lift, the required signals of the elevator can be provided via external I/O modules. These modules are connected to the display via a CAN bus wiring. This has the advantage that the modules can be installed where the signals are available, e.g. in the controller and for the connection to the display, then only the two CAN bus connections must be drawn. Thus, the wiring effort can be greatly reduced.

*The following figure* shows the basic wiring and standard configuration of an I/O module with 16 inputs/outputs.



## Commissioning of the software

### Power-up

After having set up the necessary wiring (see chapter *Required wiring*) the power supply can be connected. After about 40 seconds the display is booted and the flexyPage application will be shown on the display. Congratulations to the first successful start!

### Touchscreen calibration

If the display is equipped with a touchscreen, this can be calibrated in the menu under *Maintenance > Touchscreen calibration*. If the touchscreen is being connected to the system for the first time (after delivery), the *Touchscreen calibration* dialogue will appear automatically before the actual initialization.

## First login

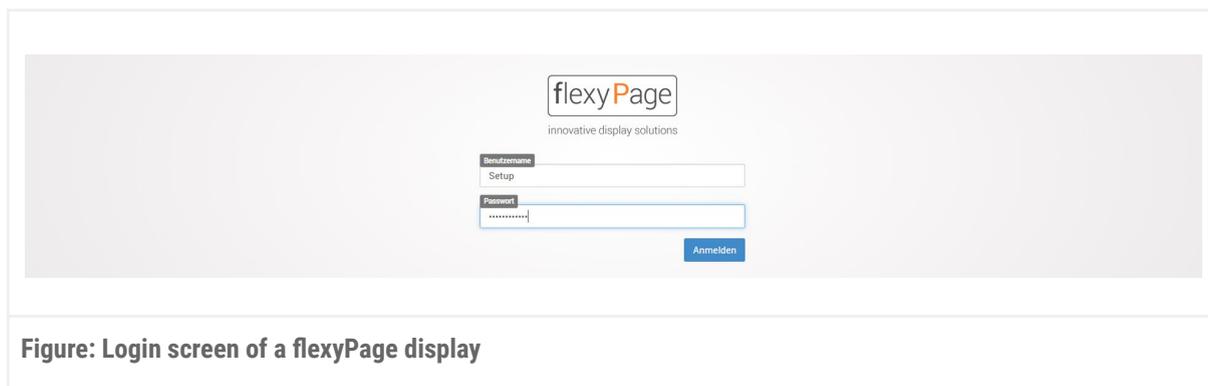
After a successful setup of the hardware, you are now able to log in into the system. For configuration you can use any up-to-date browser. The best support of the current web standards are supplied by the Chrome browser by Google. You can download it for free at:

<https://www.google.de/chrome/browser/desktop/>.

Enter the IP address of the display in the address bar of the browser. The IP address set at the delivery can be found on the sticker on the back of the display, e.g. **192.168.2.66**.

If the flexyPage device works as an access point, it is reachable via WLAN at the address **192.168.2.66**.

Afterwards, the browser should depict the login window (see following figure).



## User roles

In order to supply different user groups with setting options or to restrict their setting options, different user roles are available.

### Setup

A user with the role setup can adjust all display functions, interface parameter and the layout. This applies in particular to the technical parameters of the interfaces, e.g. the CAN parameters, IP address and date and time.

### Layout

A user with role layout is allowed to make changes to the layout and to the content. He has no access to the technical parameters, such as the interfaces.

### Content

A user with the role content is allowed to make changes to the content, which means that he can change the images of a slide show or the link of an RSS feed. However, he can not activate or deactivate content or change it in position or size. Furthermore, this user is not allowed to execute any changes to the layout or to any extended settings (coming soon).

### Update

A user with this role is allowed to install firmware updates and to load saved (layout) settings. He is not allowed to change settings or layouts.

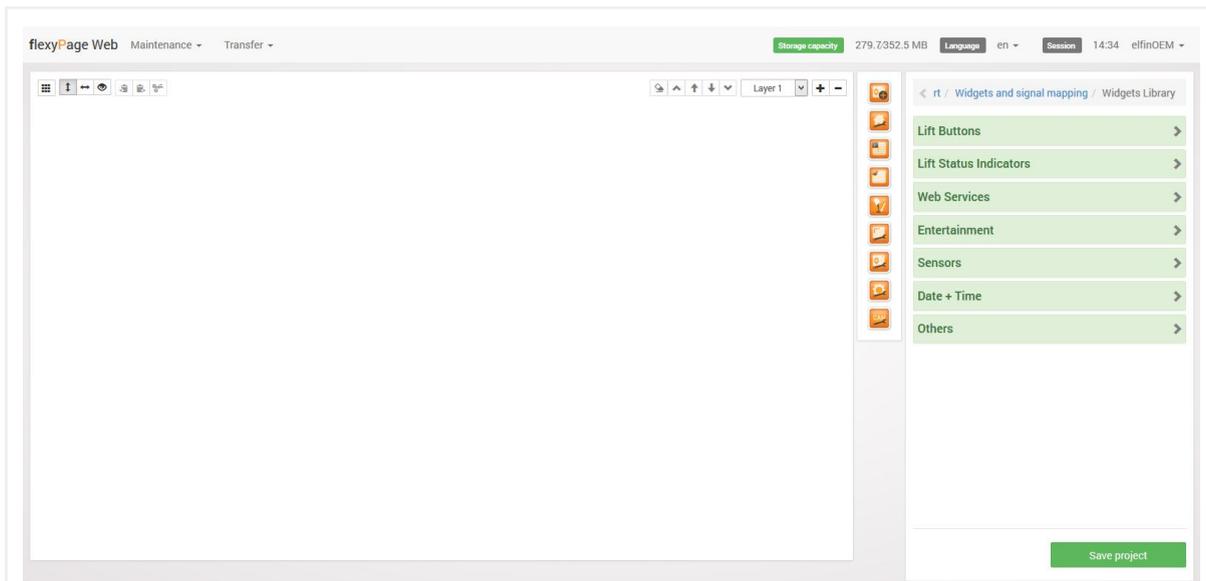
There is a hierarchical authorization relationship between the roles:

**OEM>Setup>(Content | Update | Design)**

Each user is assigned a default user role, the default view is automatically loaded when the user logs in. Additionally, every user can create representatives to which he can assign the same or a deeper role.

## Setup

After entering the access data and confirming with <Enter> or clicking on the Login-button you will reach the configuration interface of the flexyPage display (see following figure).



**Figure: Configuration interface of a flexyPage display**

## The configuration interface

The interface is divided into six parts (see following figure):

### 1. Main Menu

Here you can find the general menu items like logout and administrative functions as well as the possibility to change the user role.

### 2. Toolbar

The small buttons offer functions like toggle grid, copy, paste and cut widgets, as well as the *erase layer* function and the layer selection.

### 3. Display

In this section, the preview of the current screen configuration is displayed and can be set up.

#### 4. Quick Access

The displayed icons can be used to access frequently used menu items such as *background*, *color schemes* or the widget library.

#### 5. Menu

In this a section you can find the configuration menu. It contains both, the widget toolkit, as well as all menu items for adjusting the look and feel of the flexyPage display.

#### 6. Configuration

Via the buttons in this section you can save changes persistently or reload the last saved configuration.

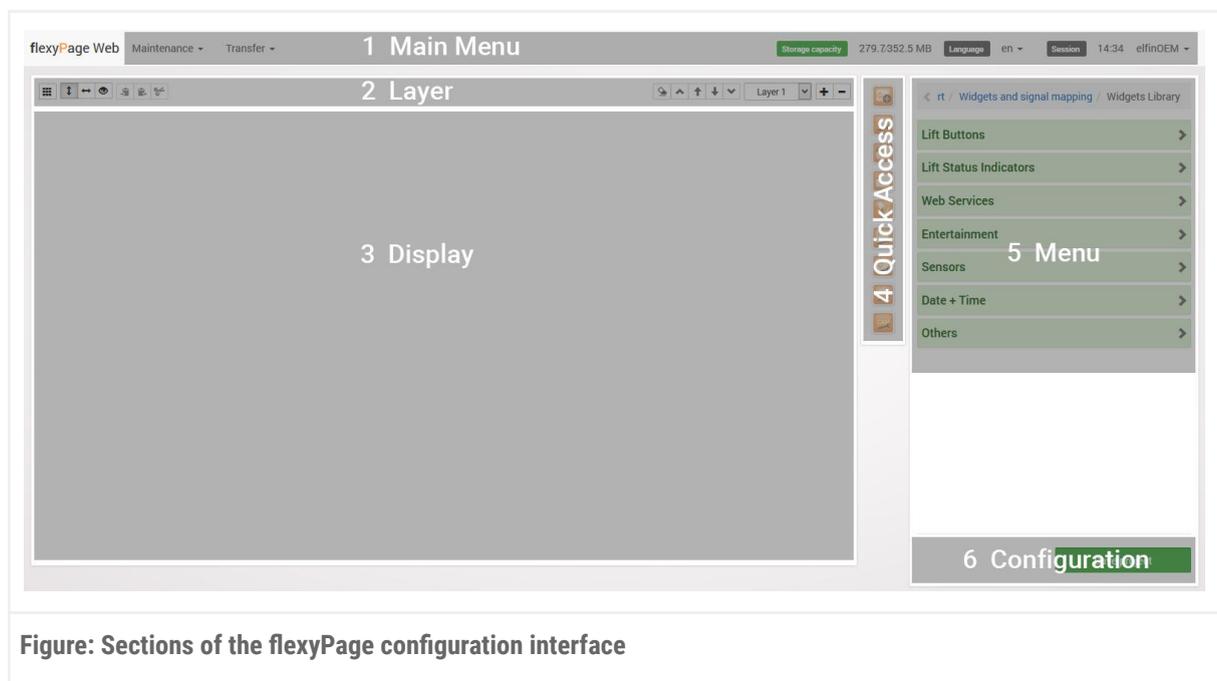


Figure: Sections of the flexyPage configuration interface

#### Shortcuts

For a quick configuration, it is possible to copy, cut and paste widgets with the usual keyboard shortcuts [Ctrl] + [c / z / v].

The current configuration can be saved and applied with the key combination [Ctrl] + [s].

#### Context menus

By clicking with the right mouse button on a widget, a context menu opens. This not only allows to copy, cut and paste the widget, but also to copy the layout and timer / event information as well as pasting

these in other widgets. The content that can be inserted depends on the two widgets. If a parameter is not available for a widget, it cannot be applied.

# First Steps

## Creating a configuration in three steps

### Layout selection

Open the layout selection via the quick access button (following figure). In the menu that will open on the right, (see figure after next) choose one of the predefined layouts by clicking on it. The predefined widgets will appear on the display (see figure on next page).



Figure: Shortcut button in the vertical quick access toolbar

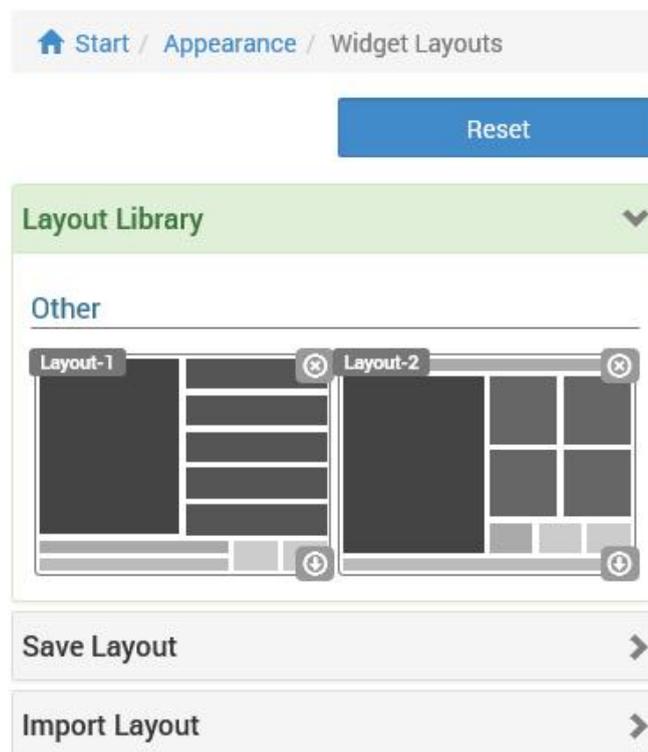
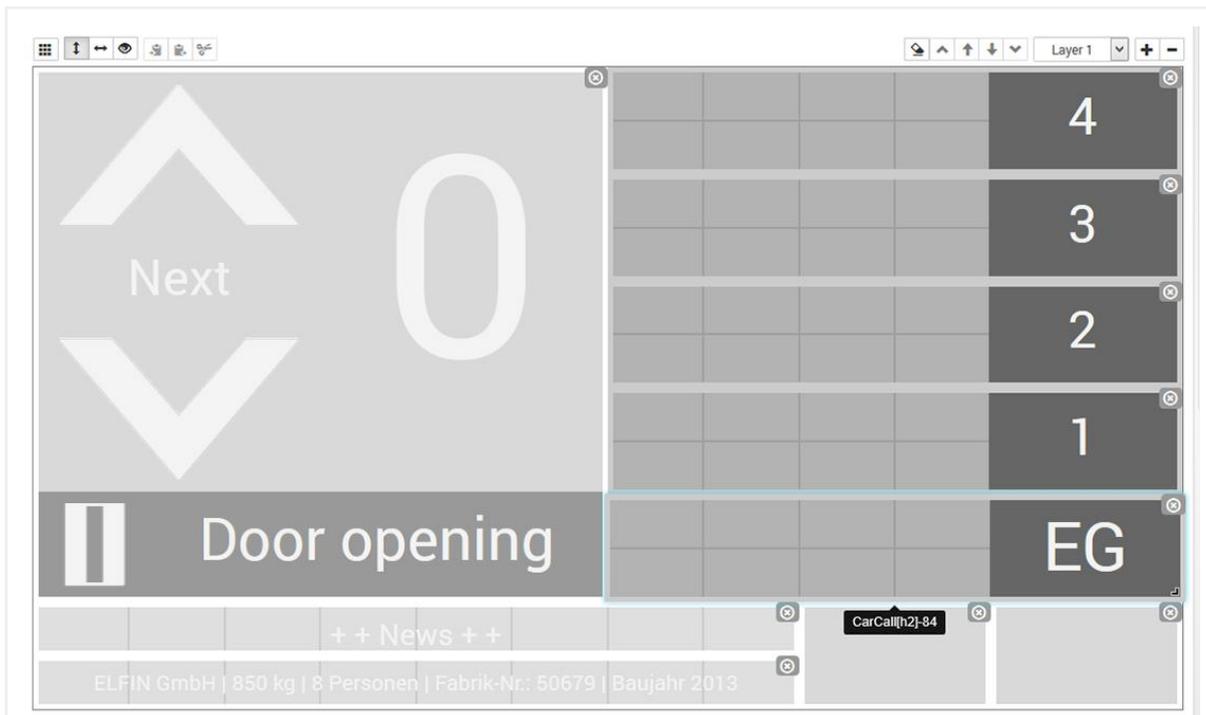


Figure: Layout selection menu



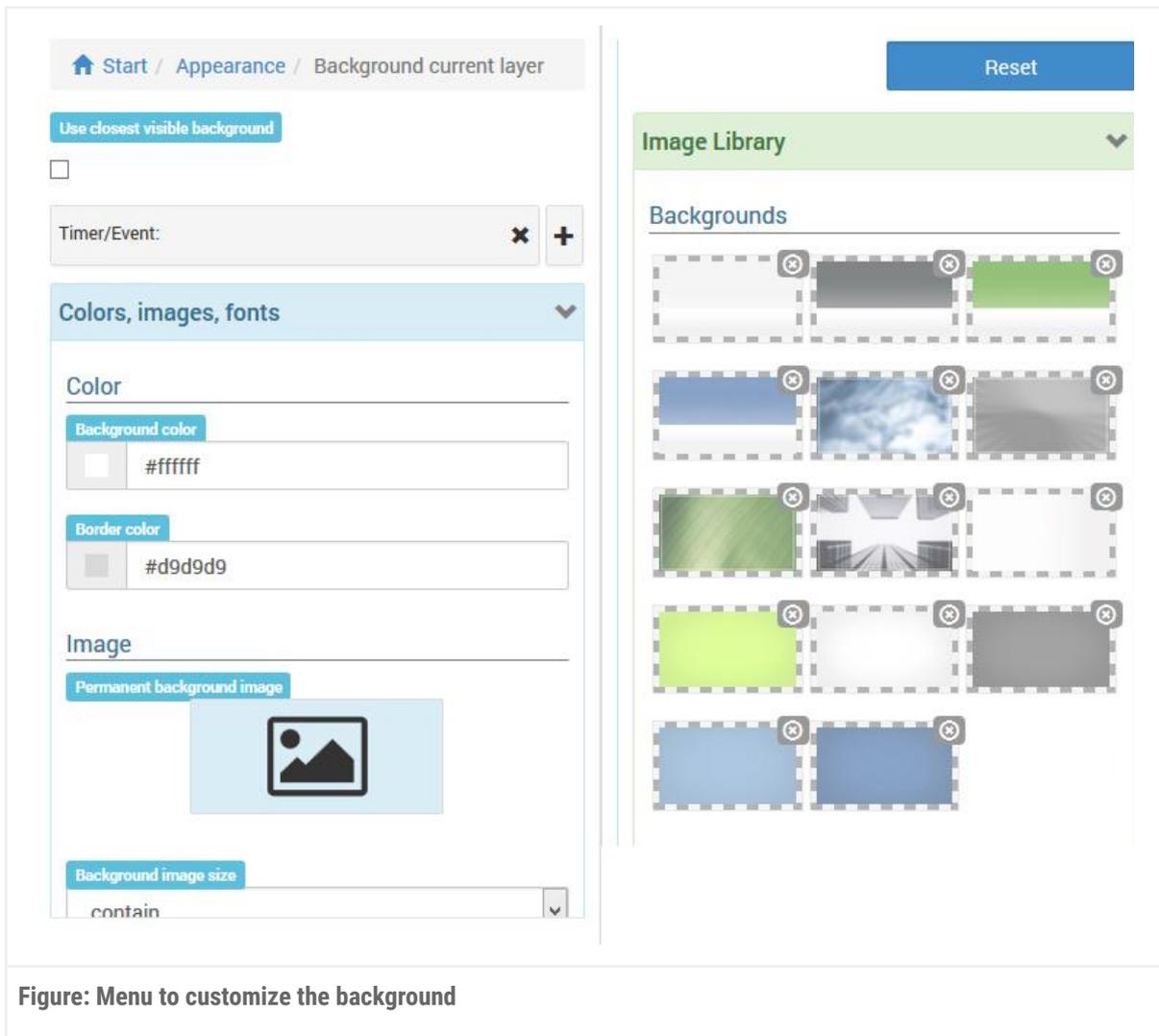
**Figure: Widget combination in the display area of the configuration interface**

## Background image

Open the background selection via the quick access button (see following figure). Choose a background in the background selection opening on the right (see figure after next) or upload a new background via the upload area and select it afterwards.



**Figure: Quick access button for the background selection**



**Figure: Menu to customize the background**

## Color scheme

Via the quick access button for the color scheme selection (following figure) you can reach the color scheme selection menu (see figure after next). Here you can choose one of the predefined color schemes or define your own five colors, from which your color scheme results, and save them.



**Figure: Quick access button for the color scheme selection**

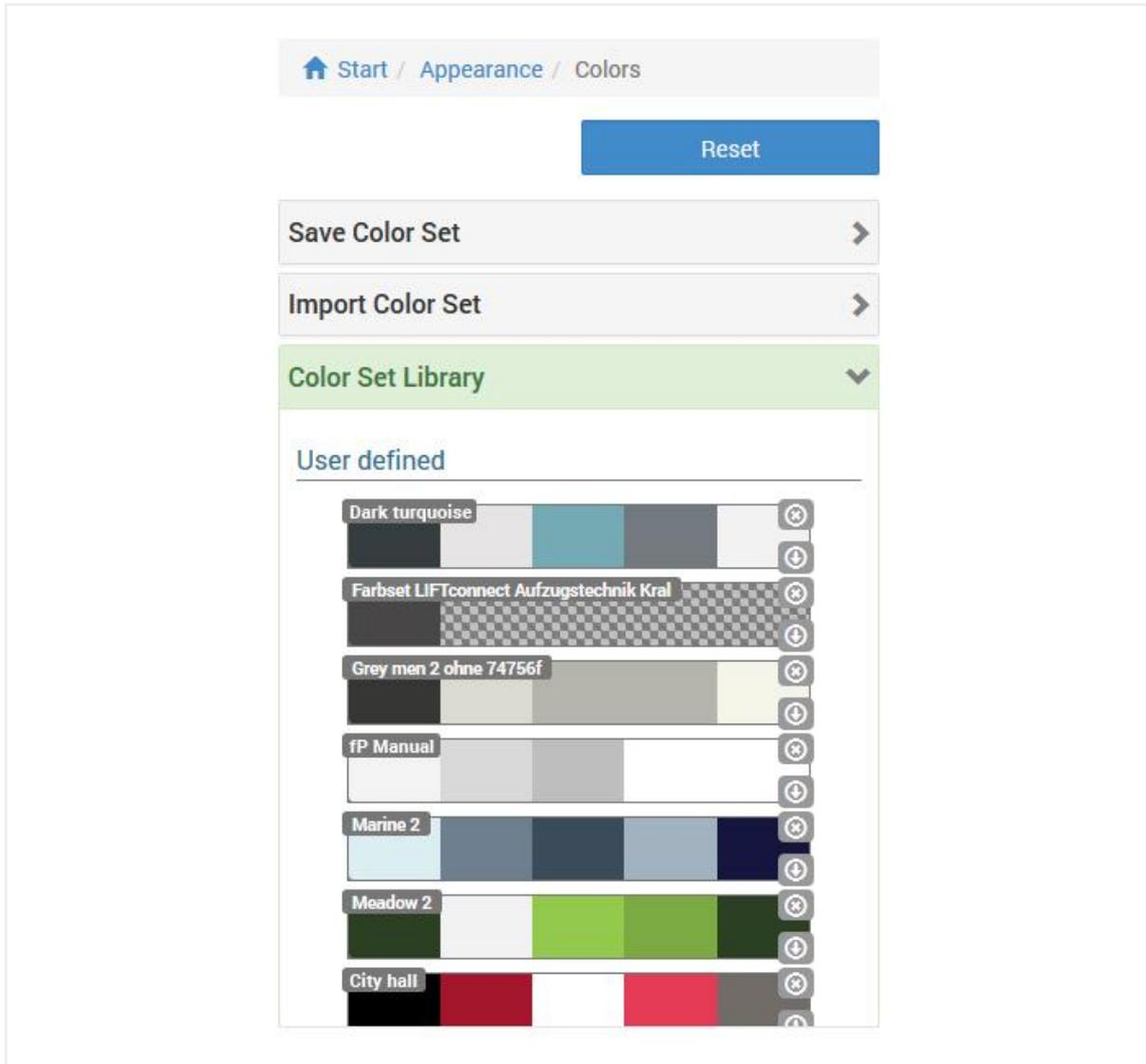


Figure: Selection menu of color schemes

Finally, the configuration you created can be saved on your flexyPage display. To do so, click on the *save button* at the right under the display area (see following figure). The changes should appear on the connected display immediately.

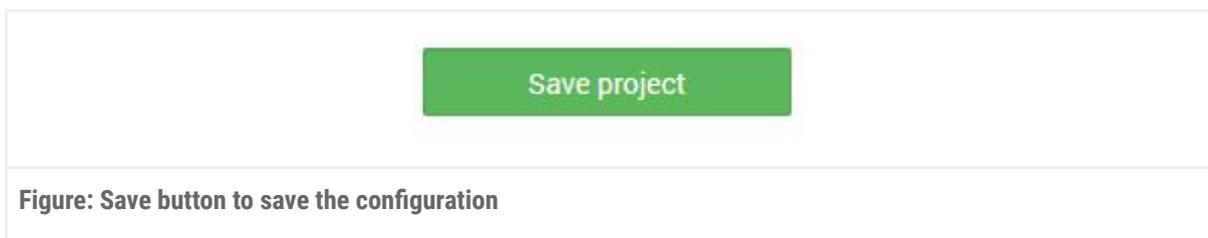


Figure: Save button to save the configuration

## Creating your own configuration

Start with an empty widget configuration. To do so, delete all existing widgets via the *delete button* in the upper right area (following figure) or use the *erase layer* button in the toolbar.

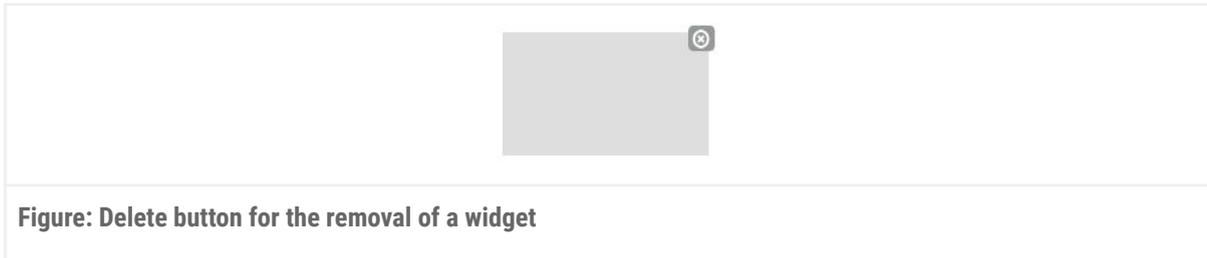


Figure: Delete button for the removal of a widget

Now, open the widget selection (following figure) via the *widget toolkit* button (see figure after next).

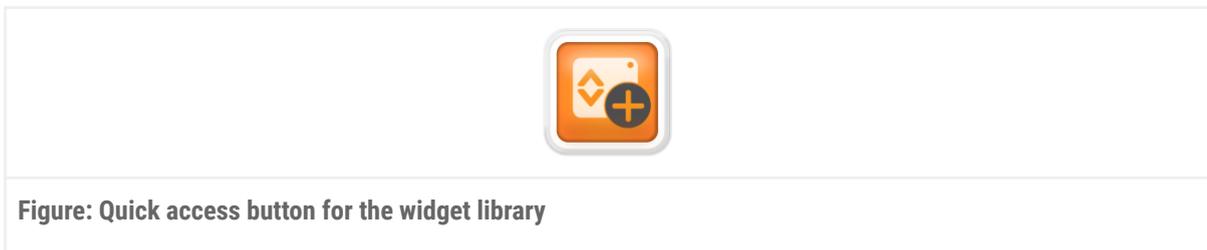


Figure: Quick access button for the widget library

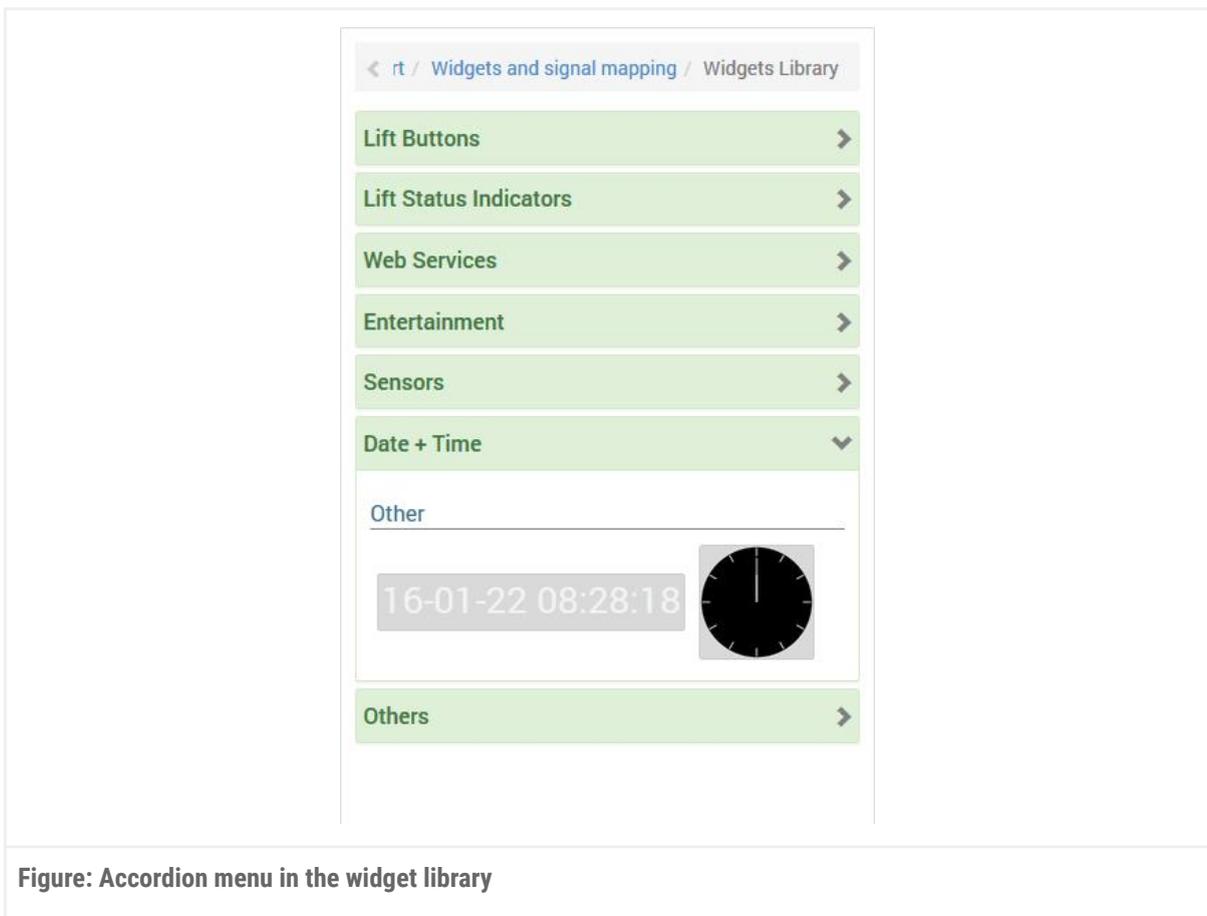
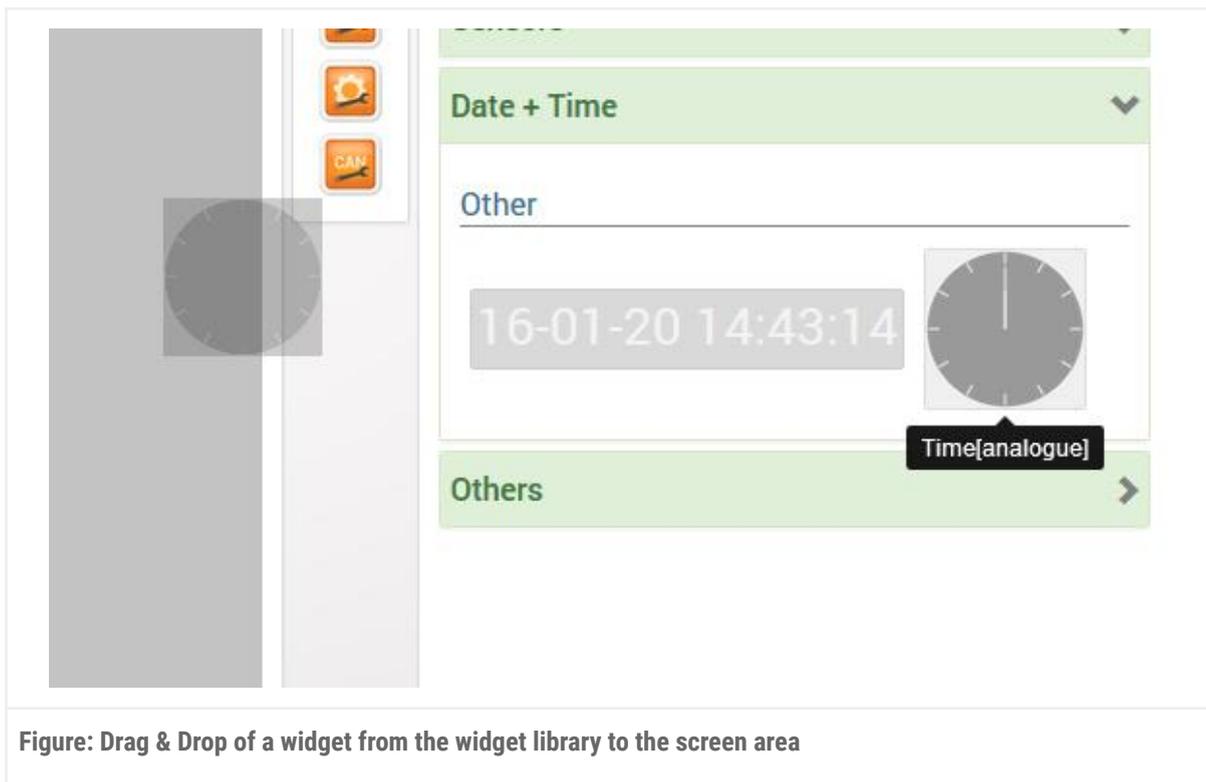


Figure: Accordion menu in the widget library

Now, you can set the desired widget combination on the display. By clicking on a widget, it will be added to the current preview. The widget will always position, as far as possible to the top left. Alternatively, you can use the drag & drop functionality to drag a widget from the toolkit towards the display (following figure).

Widgets that are dropped on the display will always position at the highest possible position, where they were dropped.



The widgets can be moved inside the display area according to your needs. The underlying algorithm ensures that the widgets do not cover each other and relocates them accordingly. A configuration, designed this way can be saved to the display with the save button down on the right side.

## Configuration of the display settings

The menu option display configuration enables the user to set the basic customization of the display design. For example, in this menu it is possible to set the arrangement of the display, the distance among the widgets or the increment of the underlying grid. The menu is accessible through a quick access button (following figure).



**Figure: Quick access button for the display settings**

## Configuration of the building

The House Editor enables an easy configuration of the tenants of a house. If a tenant moves or if his logo changes, it only needs to be adjusted in the house editor. All call buttons, floor displays and other widgets, that access this information, will adapt automatically.

It is possible to add a floor description for every floor, for example, basement or ground floor. Additionally, several tenants can be assigned to one floor and an image or logo can be chosen for every tenant. If no image is selected, the name will appear as text.

The House Editor is available via the quick access button (following figure). The menu (see figure after next) allows the easy addition of new floors and their rearrangement via drag and drop, as well as the addition of tenants to each floor. Here, the minus (-) in the header deletes an entry and the plus (+) adds an entry.



**Figure: Quick access button to the House Editor**

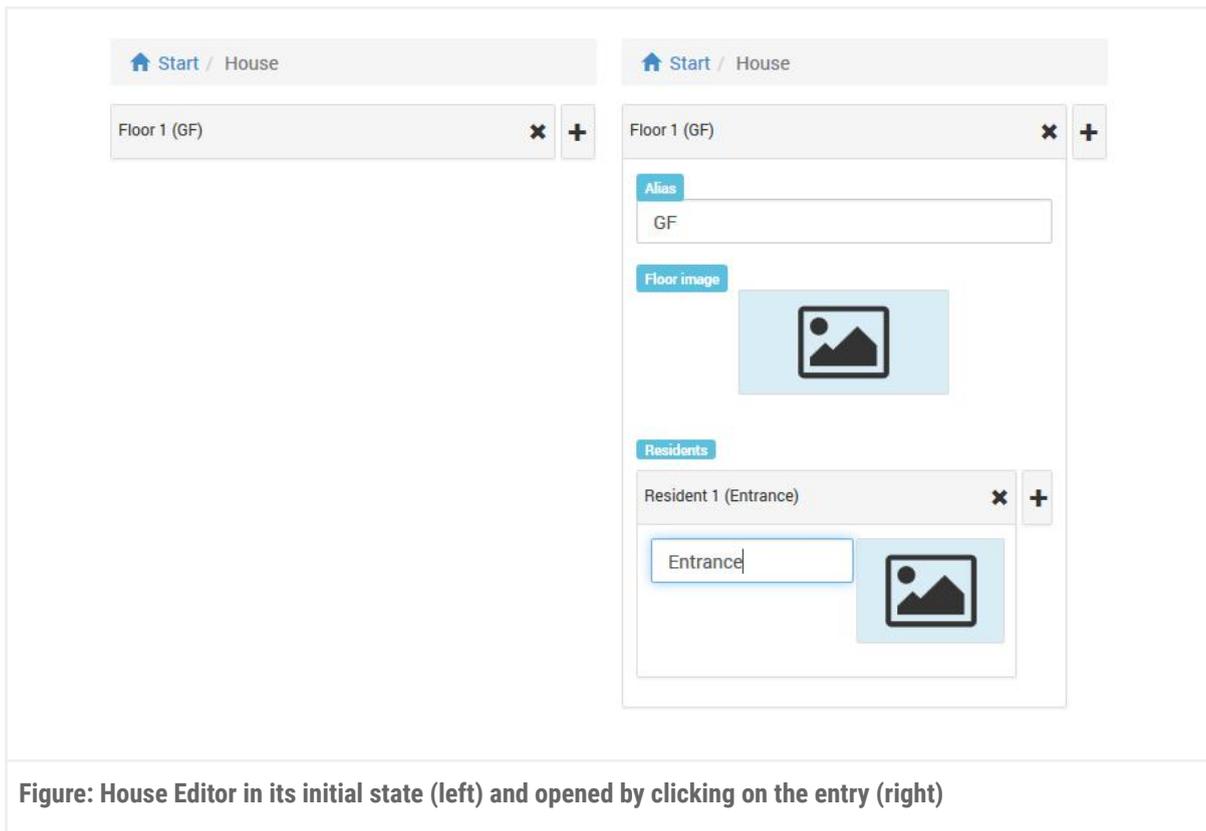


Figure: House Editor in its initial state (left) and opened by clicking on the entry (right)

### Adjusting widget properties

By clicking on the widget, the widget will be selected. The selected widget is highlighted with a light-blue border (following figure).

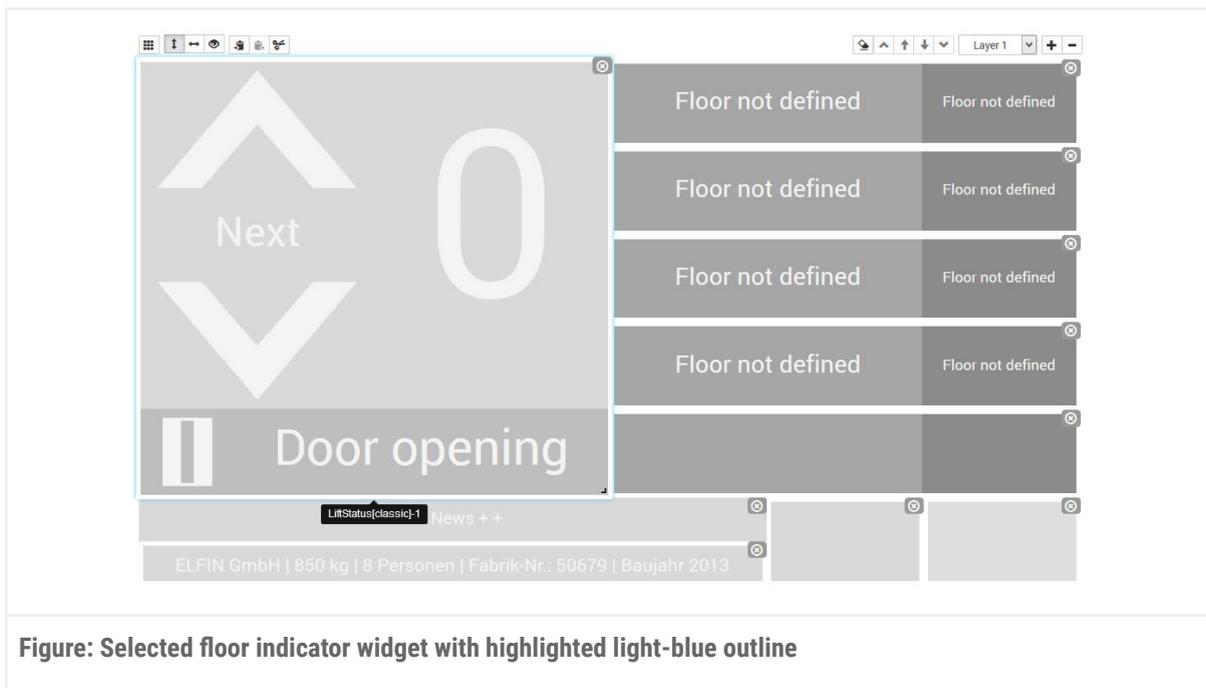


Figure: Selected floor indicator widget with highlighted light-blue outline

With the selection of a widget, a specific configuration dialogue for this widget will open on the right side (see following figure). The selection of a widget can be cancelled by clicking the widget again, selecting another widget, pressing ESC, or by a right-click.

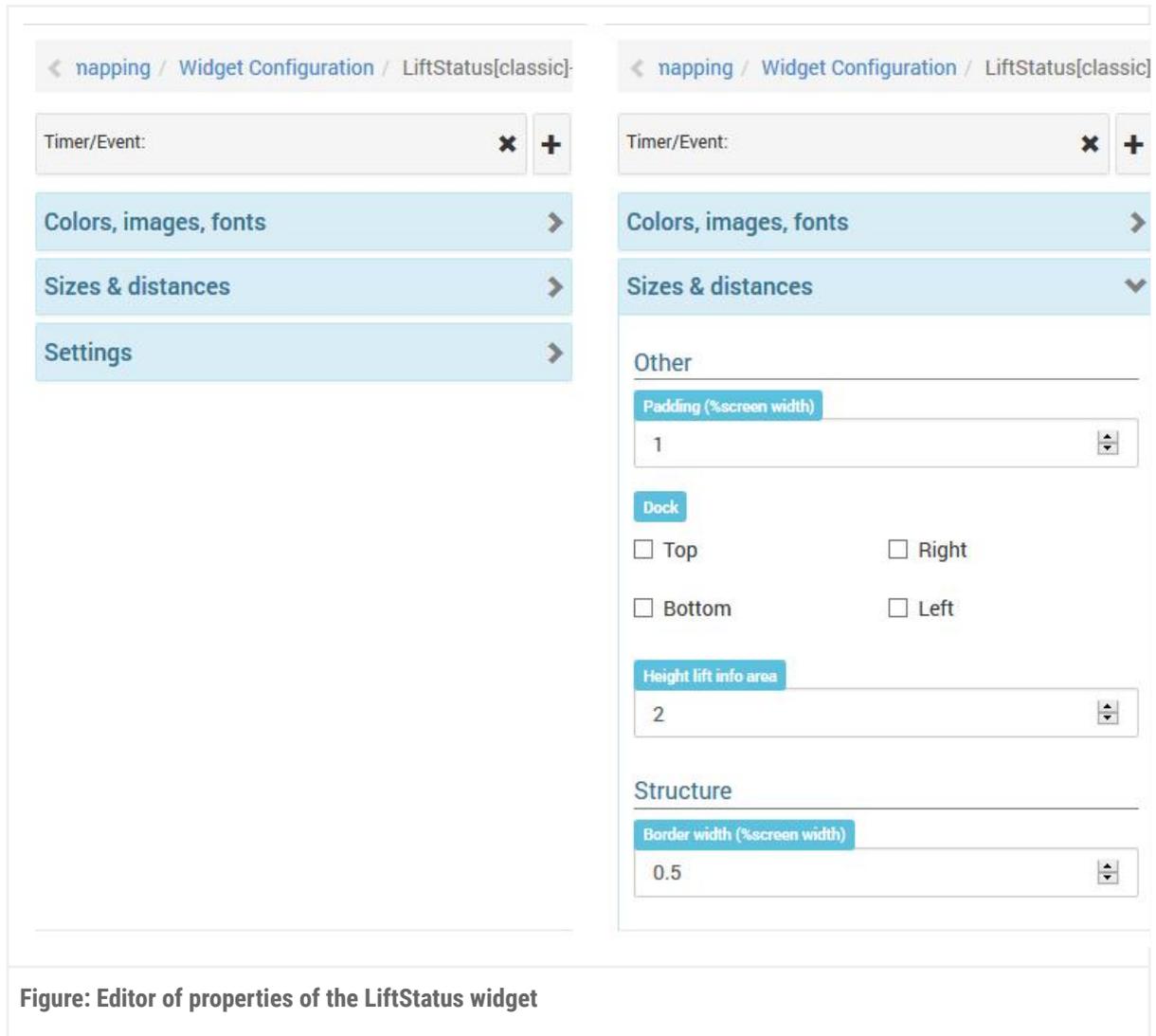


Figure: Editor of properties of the LiftStatus widget

## Device overview and connection information

An overview of the device and the current connection status can be found in the menu *Maintenance* >

*About ...*

- device name / model name
- serial number
- firmware

## Connection status

There are five connection states (online status):

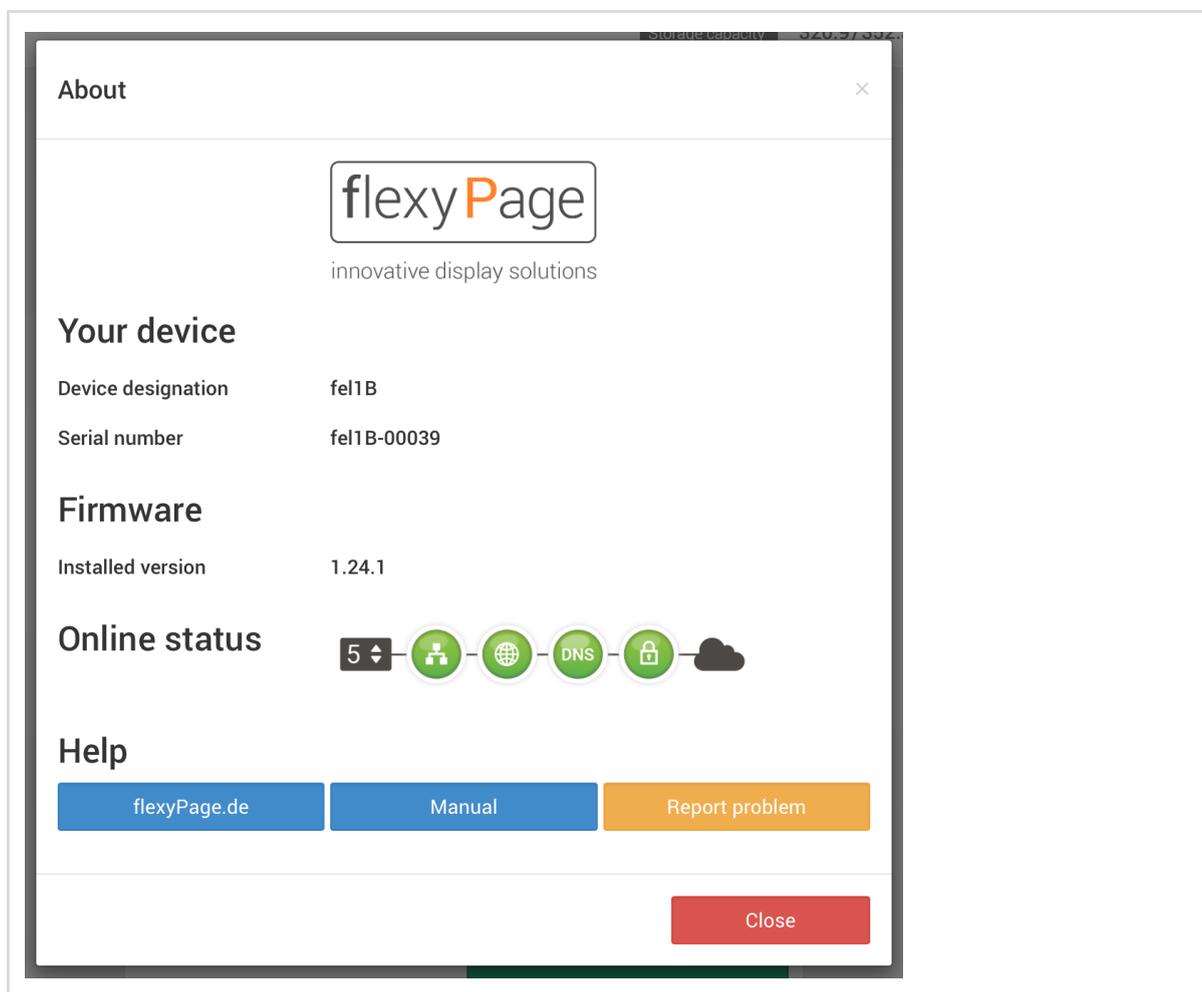


Figure: Overview of the five different connection status (online status).

### 1. No connection

The cable is not connected.



### 2. Gateway accessible

The display is connected to the gateway / router.



### 3. Internet accessible

The display can reach IP addresses on the Internet.



### 4. Name resolution possible

The display can reach a DNS server that resolves the name.



### 5. Connected to the VPN

The display can reach the VPN service flexyPage.net.



# Maintenance

The following aspects describe additional functions of the configuration interface.

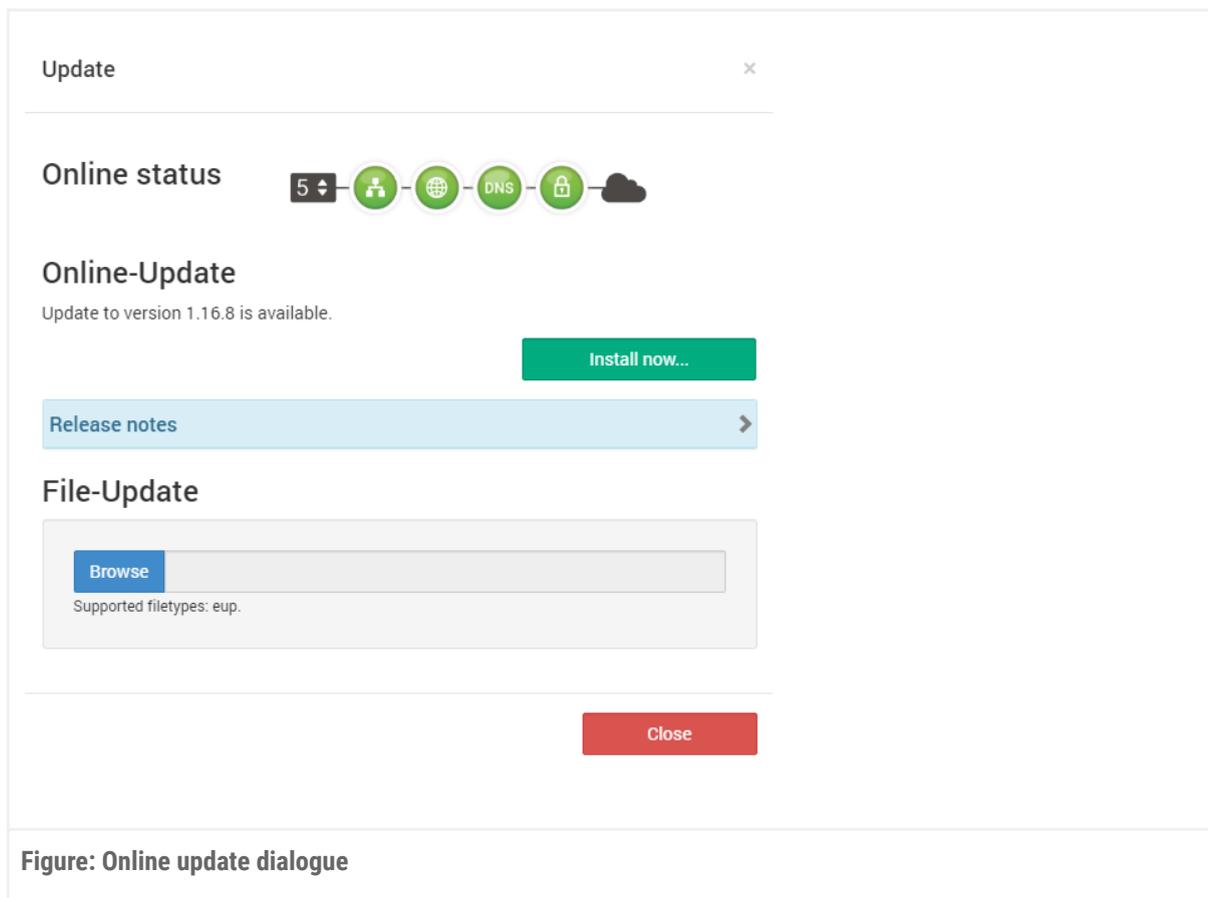
## Firmware updates

The firmware of the flexyPage displays is subject to constant maintenance and further development. On a regular basis, firmware updates are available for the displays. These offer improvements and new functions. There are three different methods available for the update.

### Online update

With an existing internet connection, the updates can be easily imported via the configuration interface of the displays. In the menu Maintenance > Update ... you will find the firmware version that is currently running on the display (under the entry Online Update) and you will get to know whether a new version is available (see following figure).

Click on *Install Now ...* to start the update process.



For the secure connection to the update server and for the remote access, a VPN tunnel is used. The tunnel uses port 2300. If this port is blocked by the firewall, the availability of updates can not be

verified, and a connection to the board via *flexyPage.net*, is not possible. Consequently, internet access will not be possible and the green indicator with the safety lock, which is visible in figure before, will turn grey.

## USB-Update

It is also possible to run the updates via an USB flash drive.

### Preparing the USB-flash-drive

The installation via the USB flash drive requires a FAT32 formatted USB flash drive. Please note that a partition is created on the stick. Without this partition, the update will fail. The previously downloaded file is then copied onto the prepared stick. The file must be at the top level of the directory structure. The file name needs to be in the form of:

update-<type><version>.eup

thus, for example

update-fel1A1.16.4.eup

If a version <0.22.x is installed on the display, the file on the USB flash drive must be renamed to *update.eup*.

### Update

The prepared USB flash drive can now be plugged into the USB slot of the flexyPage controller. After restarting the board, the update is installed. This process may take several minutes. Finally, you should be able to read the updated software version on the start screen of the display application. If the <version> does not appear in the version field, the update has not been successful. In this case, the alternative described below can be used. After the configured user interface is visible again, the USB flash drive can be removed.



**Make sure that the configured display surface is displayed and that the applications that are running on the board are started, before you remove the USB flash drive.**



**Note: Up to software version 0.22.x, USB updates did not check whether it is an update file for a fel1A or fel1B device. The files may not be interchanged, otherwise it may happen that a LVDS connected screen remains black.**

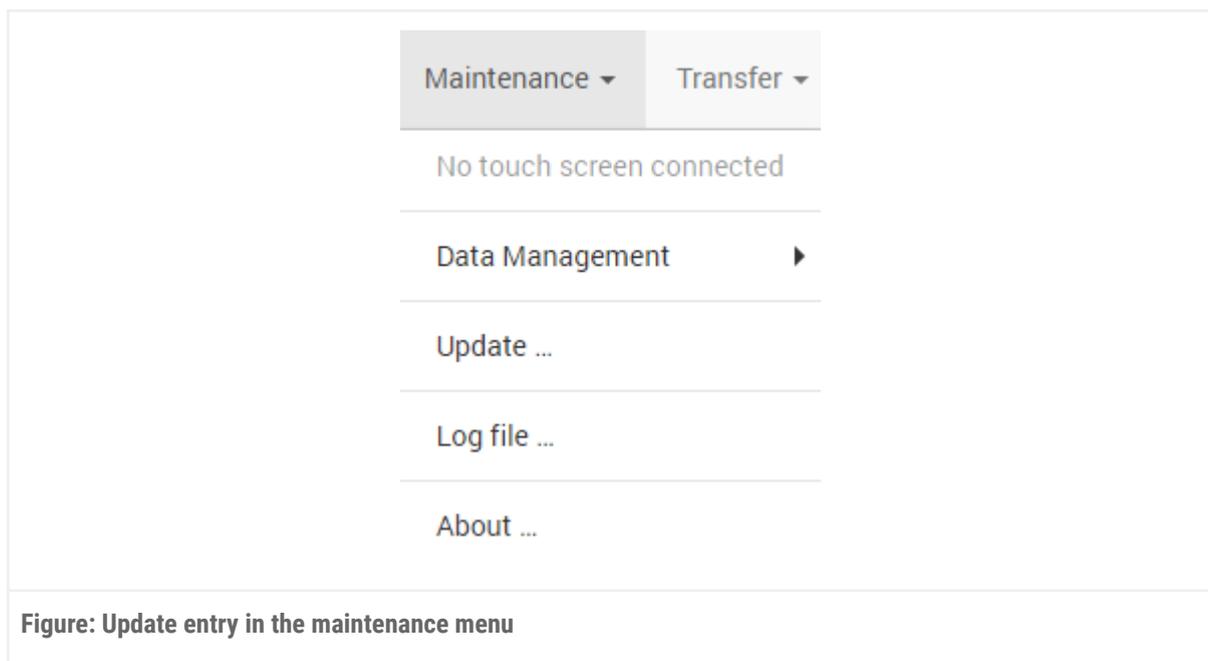
For some older firmware versions, it is not possible to install the latest version directly. The updates must be installed in the following order:

- FPC01: 0.11.0 > 0.12.0 > 0.18.0 > 0.22.0 > 1.22.1 > latest firmware
- FPC02: 0.22.0 > 1.29.0 > latest firmware
- FPC04: No dependencies
- FPC05: No dependencies

For updates via the flexyPage Editor, the correct order is automatically selected.

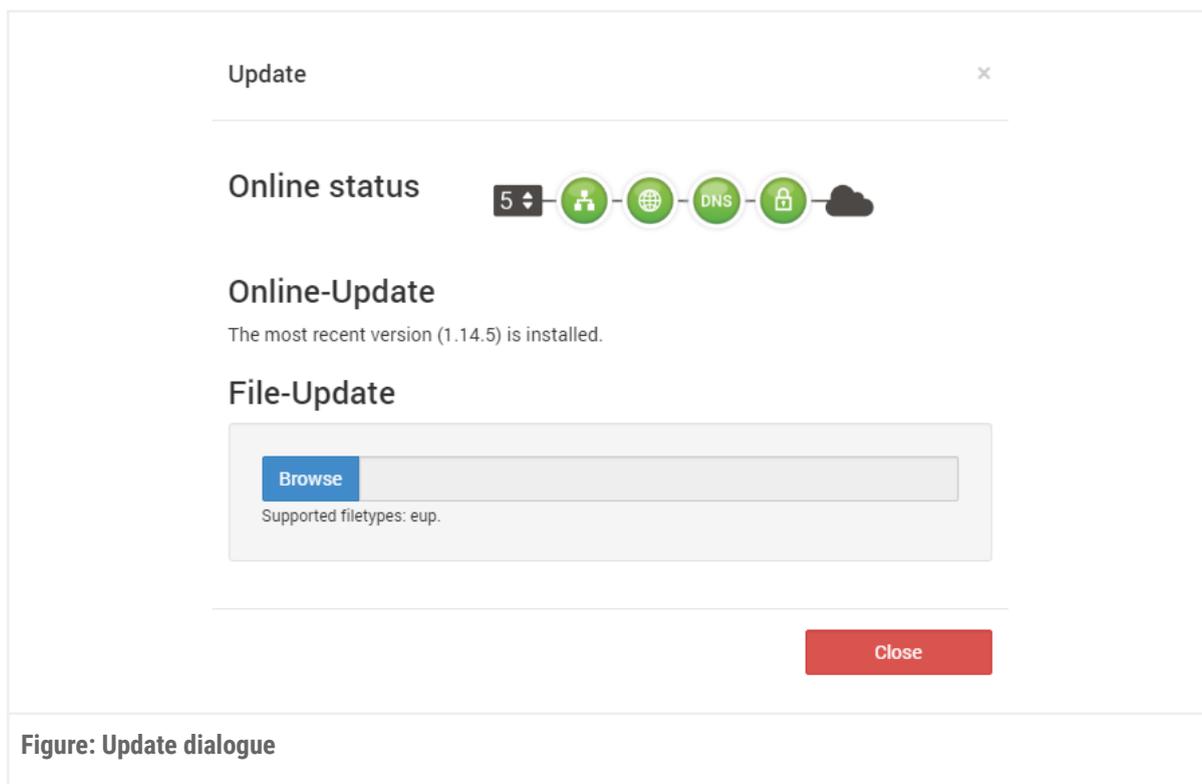
## Upload a firmware file

The flexyPage Editor provides an update dialogue. This can be accessed via the *Update* menu item in the *Maintenance* menu (see following figure). To do so, log into the flexyPage Editor.



## Update dialogue

The Update dialogue (see following figure) provides a dialogue *File-Update*, that contains a field for the File Upload.



The update file downloaded from *flexyPage.net* can be uploaded here. After upload, follow the dialogues that appear in the flexyPage Editor. You will be informed about the update status via a progress bar. Finally the device is restarted.

## Calibration of the touchscreen

For the calibration of the touch screen see *Comissioning > Comissioning of the software > [Touchscreen Calibration](#)*.

# Hardware settings

## Localization and time

### Time

The current date and time of the display can be configured.

If a very precise time is required, we recommend you to obtain the time via the NTP protocol from the Internet.

With some models, in the event of a power failure, the clock is buffered with one battery, with others for a duration of about 48 hours by means of a gold cap.

### Area/Time zone

For the correct representation of the date and time as well as the change between summer and winter time, the appropriate place must be selected.

### Language

This parameter sets the display language. It is not the language of the configuration interface but the content of the display, such as the special texts.

## Screen

If your flexyPage device is connected to a monitor via HDMI, you can select the resolution. The available resolutions depend on the connected monitor. We recommend you to set the correct resolution instead of using an automatic detection.

## Network settings

A fundamental feature of the flexyPage displays is the integration of up-to-date Internet technologies for the configuration of displays, for displaying web services (e.g. current news), as well as for remote diagnostics. To use of these functions, the integration of the display into a network is necessary, via which access to the configuration interface can take place, or via which a connection to the Internet can take place.

### Network settings of the flexyPage display

flexyPage displays are delivered with factory-set network settings. This allows direct access to the configuration interface with a PC. You can adapt the network settings to your needs and circumstances. For this you should have basic knowledge of network technology.

## LAN / IPv4 settings in the flexyPage display

### Factory settings

The following IPv4 settings are factory-set in the flexyPage displays:

Settings	Preset value
IPv4 address	192.168.2.66
subnet mask	255.255.255.0
DHCP server	off

From IP address and subnet mask the following IP addresses can be derived:

Settings	Value
network address	192.168.2.0
complete IPv4 address range for the computers	192.168.2.1-192.168.2.65 and 192.168.2.67-192.168.2.254

### Reserved IP addresses

The following IPv4 addresses are reserved to certain purposes and thus can't be assigned:

IP address / designated use	Value
network address	192.168.2.0
used by flexyPage by default	192.168.2.66
used by display for remote access	10.0.0.1

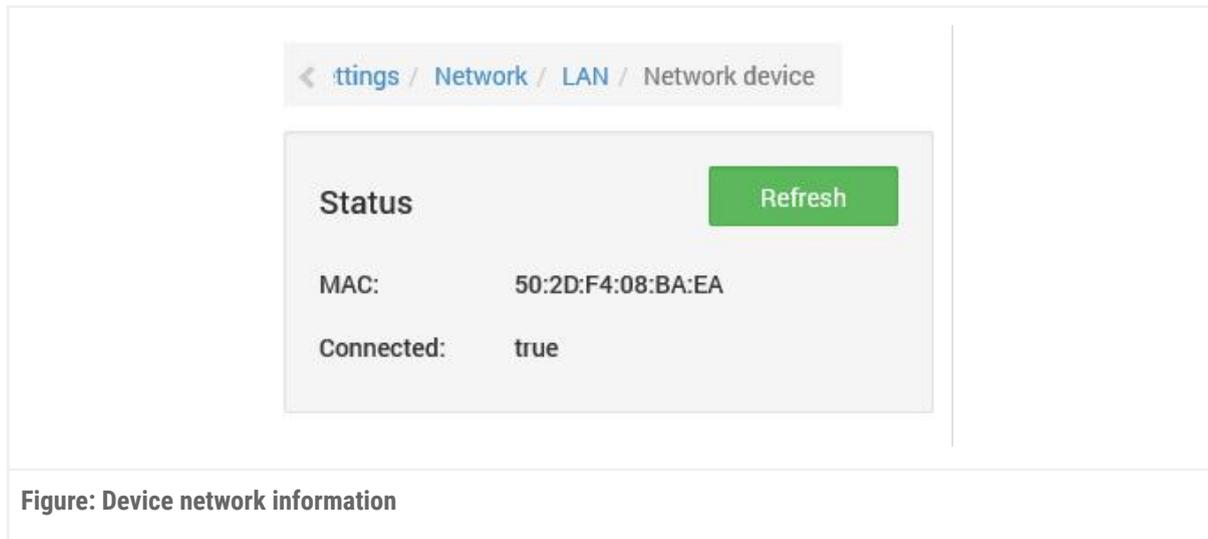
### Changing the IPv4 address

In case of the following circumstances the adaption of the IPv4 settings is useful or necessary:

- You have an existing local IPv4 network with multiple PC's over which the access to the display is supposed to take place.
- You have an existing local IPv4 network over which the internet connection is supposed to be provided.
- You have an existing network in which the IP addresses for the participants are supposed to or necessary to be given dynamically.
- You want to connect several flexyPage displays together.

To change the network settings of the display, log into the display (see chapter *Connection to the internet via web browser*). In order to change the network settings, you need the rights of the user role *Setup*. You can find the network parameters of the LAN interface under: *Hardware settings > network > LAN > network device*.

In the upper area of the dialogue, the current status of the network adapter is displayed. Here you will find the MAC address and you can check whether the adapter is connected. Below, the settings of the active adapters are displayed. With a click on this overview, a dialogue in which the parameters can be adjusted, opens (see figure below).



Enter the appropriate settings for your network here. If you have any questions about these parameters, please contact your network administrator.

If you want to connect directly to your computer with the display, you must adjust the network settings of your computer.

Information on this can be found in our video tutorials

(Windows):

<https://flexypage.de/en/how-connect-display-ethernet-your-windows-personal-computer>

(Mac): <https://flexypage.de/en/how-connect-display-ethernet-your-mac>

or the instructions under this external link:

<https://www.heise.de/tipps-tricks/IP-Adresse-unter-Windows-10-aendern-so-geht-s-4267953.html>

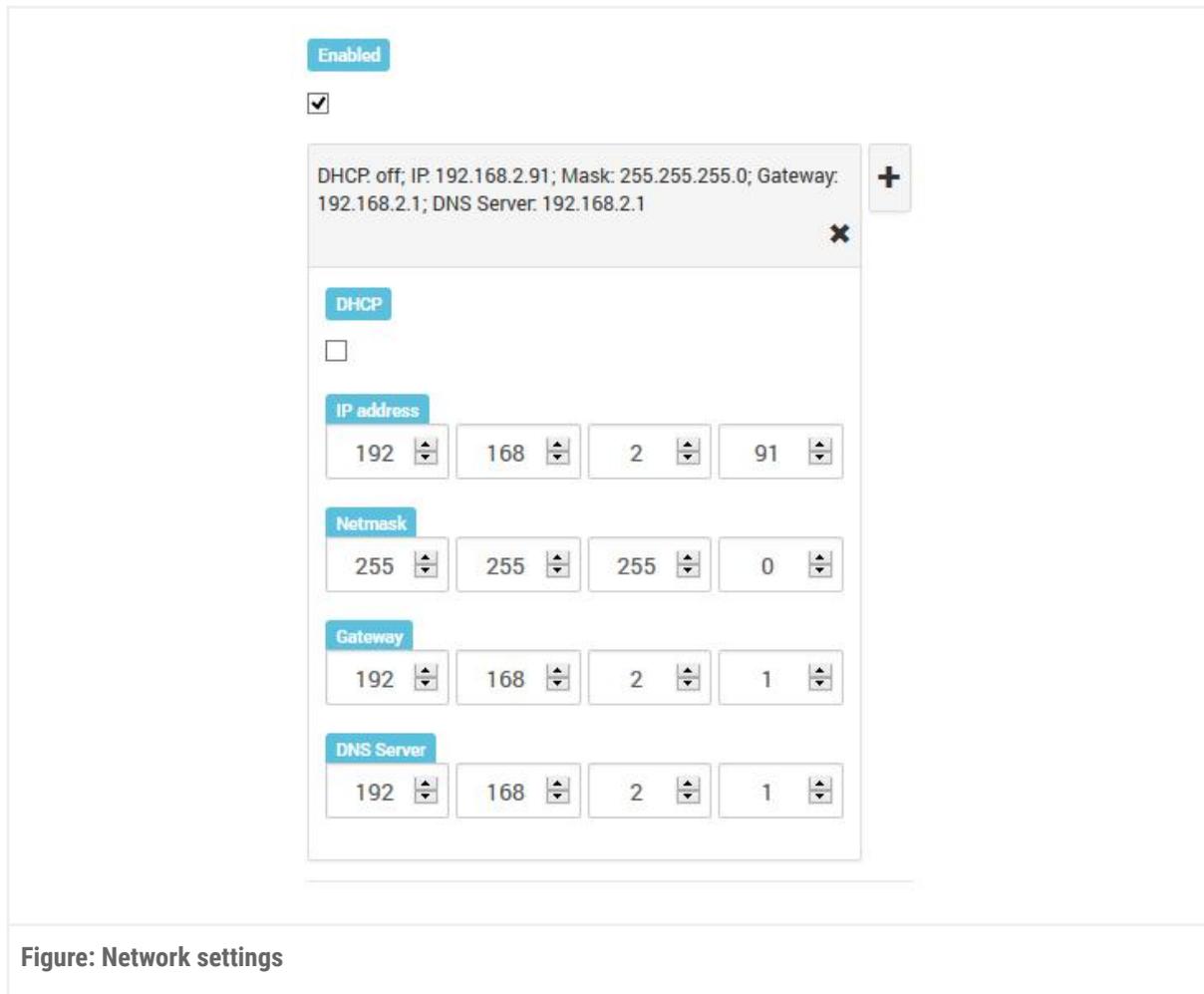


Figure: Network settings

### Displaying the network settings in an emergency

If the network settings have been changed after delivery, access to the configuration interface in the local network is only possible via the newly set, or the dynamically assigned IP address, from a DHCP server.

If the flexyPage display has internet access and if you have assigned the display to your flexyPage portal, you can always access the configuration interface via the portal.

If the display is not connected to the internet and you do not know the current IP address, you have the following options to determine it:

- In case the display is integrated in a network, please turn to your network administrator. He can provide information about how to get access to the display.
- If the info screen is activated in the settings, a pop-up window will display the network settings, shortly after turning on the display.
- Via the flexyPage support stick, the output of the IP settings can be forced or the display can be set back to factory settings. Please refer to the support.

## Cellular (4G)

flexyPage displays are optionally available with an integrated 4G modem.

- **Enabled:** Switching the 4G interface on/off. When the 4G interface is active, network traffic is primarily routed via this interface. However, the configuration interface is still accessible via LAN under the configured IP and local data sources, e.g. the stream from a network camera, are still accessible.
- **APN:** Access data for your mobile network. You will receive these data together with your SIM card from your wireless carrier
- **User:** Access data for your mobile network. Most providers leave this field empty. You can obtain this data together with your SIM card from your wireless carrier
- **Password:** Access data for your mobile network. Most providers leave this field empty. You can obtain this data together with your SIM card from your wireless carrier
- **PIN:** PIN code of your SIM card. You can obtain this data together with the SIM card from your wireless carrier

If your SIM card requires the PUK, it is necessary to insert the card into a device with buttons (e.g. mobile phone) and enter the PUK there.

### **Mobilcom-Debitel (Vodafone Netz)**

- **APN:** web.vodafone.de
- **User:** vodafone
- **Password:** vodafone

### **LIDL Connect**

- **APN:** web.vodafone.de
- **User:** [EMPTY]
- **Password:** [EMPTY]

### **Telekom M2M**

- **APN:** internet.t-mobile.de
- **User:** [EMPTY]

- Password: [EMPTY]

## Network services

### NTP Settings

The system time can be obtained from a local or in the internet situated NTP server. The server data can be entered under *Hardware settings > Network > NTPServerSettings*. Port 123 has to be open for query of the NTP server.



**Attention!**

If a Proxy Server is used, it is not possible to use an external (internet) time server. In this case only the use of a network-internal time server is possible!

### Proxy server settings

The Internet access via a proxy server is possible. At the moment the following protocols are supported:

- HTTP

The proxy data can be entered via *Hardware Setting > Network > Proxy ServerSettings*.



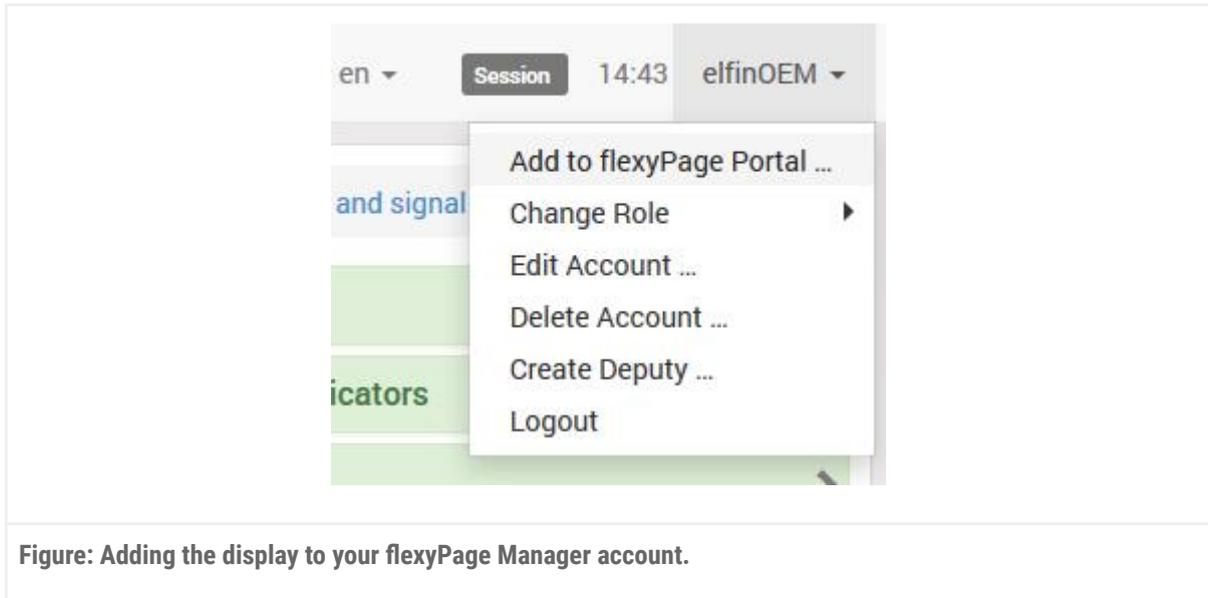
**Attention!**

An online update and the remote access is not possible with an activated proxy server!

## Remote diagnosis

In the case of an existing Internet connection of the flexyPage display, remote access to the configuration interface is possible in principle. An account on the *flexypage.net* configuration portal is required for use.

If remote maintenance via this account is possible, the flexyPage device must be assigned to this account. This is possible via the menu item *Account name (right) > Add to flexyPage Portal ...* (see following figure).



**Figure:** Adding the display to your flexyPage Manager account.

By clicking on this menu entry, a redirection to flexypage.net is made, in which information about the identification of the current device is sent to the flexyPage Manager. After a successful login, a mask for filling in further information opens. A short note, a detailed text, which can only be seen by a logged-in user and the device's actual position, can be given (following figure). While the actual position refers to the display, this information is adapted for all users, who have access to this device.

A transfer back to the device will be carried out after successful saving.

## Add new device

**General information**

**Serial number**  
fel1A-00007

**Short note**  
Enter some short description

**Full note**  
Enter a useful description to identify this device

**Location**

**Address \***  
Geben Sie einen Standort ein.

Save

**Figure: Input mask for saving identified device information and ascribing to the Manager account.**

## Export/Import

There are several export possibilities to exchange data with several displays or to have a data exchange for maintenance purposes. The transfer export enables the exchange of settings between several displays. Via the menu entry *Transfer > Export* an encrypted data archive can be exported, which can be imported again on other displays or the same display via the menu entry *Transfer > Import* (following figure).

An export includes all information to rebuild the same condition on another device. During the import it can be chosen which settings should be adopted.



### Attention!

The import will transcribe the current settings.

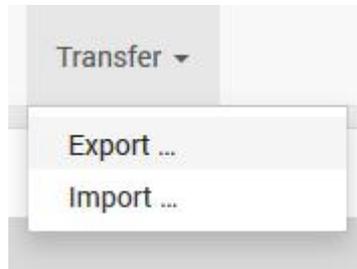


Figure: Menu entry for the transfer of data. In *Export*, you can export a signed archive which can be imported again via the *Import* function.

## General display settings

### Distance between widgets

Widgets are aligned on a grid. It is possible to set a general distance to be kept between all widgets. This is set in percentage of the width of the screen.

### Display rotation

The display content can be rotated by 90°, 180° or 270°. This allows using the display in portrait or landscape mode. Another advantage is, that during the installation no attention has to be given to the orientation of the display, because the content can be rotated afterwards. Also note that when rotating the content, it can be stretched or squeezed, so the choice of matching images when creating the layout, is of high importance.

### Grid units

The number of grid units on which the widgets are aligned can be set to any number between 12 to 48 on the x as well as y axis. This allows very exact positioning of widgets, which is especially important on large screens or when many widgets are being used.

## System information at start of programme

When this option is activated, there will be a pop-up window displayed for a few seconds after every starting of the display. This depicts the serial number, current network settings and the current software version. In an emergency this information helps with a quicker diagnosis of problems.

## Special Text

### General

In the application profile CANopen CiA417, the special signals (messages) that can be output on displays or in displays, are defined. Here, both, controller and other bus users, can output several special signals simultaneously. Some special signals are only informative, while others are required by the norm. The setting of the special signals for the respective application takes place by configuring the displays. Depending on the display, different special signals can be displayed at the same time, whereby some of the special signals defined in the standard should only be output with special applications. In addition, flexyPage displays can display additional data as special text, such as the status of special signals and the status of the door.

Out of this, the following categories result:

- **Special signals**

In this menu, an image and a text can be defined for the special signal message. These will be used by all widgets that can display information.

- **Special functions**

In this menu, texts and pictures for special functions can be defined.

- **Door**

An image and a text for the states of the door are determined here as well. Again, these will then be used by all the widgets that can display this information. However, in order to be able to display these states, a CANopen-capable door is necessary.

### Deactivation of special signals or functions

If a message shall not be displayed and shall be ignored by the widgets, the text field needs to be empty. Each lift controller sends different signals. Therefore, the parameters have to be adapted for each of them.

## Priorities

The special signals have first priority among messages. Additionally, there are priorities among the special signals. The following signals are shown according to their priority when they are active. In case of a single-line display, only the message with the highest priority will be shown.

### Signals per priority

- |                       |   |                      |
|-----------------------|---|----------------------|
| 1. (highest priority) | - | Overload             |
| 2.                    | - | Fire brigade service |
| 3.                    | - | Fire/Case of fire    |
| 4.                    | - | Special service      |
| 5. (lowest priority)  | - | Out of order         |

With the signals of *Fire brigade service* and *fire/case of fire*, the message *out of order* will also be sent through the controller. All other signals are considered *informative signals*, which can also be displayed, as long as no higher priority signal is present, and/or sufficient area is available in the displaying widget.

Some widgets allow multiline information display, to show multiple signals simultaneously.

The signals with high priority are shown first, while additional informative signals are shown in the other lines available. We also recommend to disable by default the following signals since they are only needed in special cases:

- *Besetzt/Occupied*
- *Türe schließen/Close Door*

All other signals will follow the priority sequence of their CAN-ID.

Special functions will be shown after special signals. The door status is shown in the end if the door is CANopen compatible.

## CAN Configuration

There are two options to connect the flexyPage display to the lift controller:

- via CAN with a CANopen-Lift compatible lift controller.
- via the in-/output of a controller, using an additional CANopen in-/output board.

If the display is connected with a CANopen compatible controller, this will perform like a common CANopen compatible display or keypad (Input panel unit/output panel unit). If the controller is not compatible with CANopen, it is necessary to use a CANopen Lift in-/output board. The board must be then connected to the lift controller so as with the CAN bus of the flexyPage system.

The display controls the CANopen master. It is assumed that the master has a node ID in the range 1 to 8. If the node ID is not available, the mode *Out of service* will be activated in the master.

## Lift configuration

### Simulation

There is a simulation modus for demonstration purposes, which emulates a lift drive between 6 floors. During the simulation the CAN interface is turned off and it does not react to inputs on the display.

### Baud rate

The speed of the CAN interface. Currently baud rates of 250 kbit/s and 125 kbit/s are supported. As most of the elevator systems use a baud rate of 250 kbit/s. This is the standard value. If the given baud rates is wrong, the display can't communicate with the lift controller and therefore nothing will be shown on the screen.

### Nodes-ID

The CANopen Node-ID of the device is default for 125 configured. In the CANopen standard it is recommended to set up a node-ID from 16 to 20 when installed in an elevator cabin and a node-ID between 21 and 85 for installations in the shaft.

### Floor

The *floor* refers to the floor on which the display is installed. This parameter is necessary for the display to communicate with the lift controller successfully. If the display is installed in the elevator cabin, "0" should be set up in this field.

### Assigned lifts

If the display is connected to a group of lifts, it must be indicated from which lifts the display should display information. If there is only one lift or a lift system independent from others, number "1" should be selected.

## Door

Here it can be indicated to which door the lift is assigned to. It is possible to assign up to 4 doors to a single display.

## Load sensor settings

flexyPage displays can assess the information of a CANopen-Lift load sensor and show it on the screen, using a widget (see *Widget > Sensors*). To enable the display to show correct data, some information of the lift must be set:

- **Overload**  
The overload limit of the lift in kg.
- **Full load**  
The full load limit of the lift in kg.
- **Minimum load**  
The minimum load of the lift in kg.
- **Tara**  
The tara value of the lift car, if this is not automatically given by the sensor.

## Position settings

The flexyPage display can use the data of the elevator car position sensor and generate messages. Therefore, it is necessary to set some parameter of the sensor.

- **APS-Resolution**  
The number of measure points per revolution.
- **APS-Length**  
The length of the circumference of the disc or the distance in millimeters to which the disc has moved with reference to the preset measurement points.
- **APS-Offset**  
The offset of the position sensor in millimeters that arises due to the existing pit and because the calibration was not set to zero. The value will be automatically set after the learning drive.
- **Floor positions**  
Learning drive

- In the learning drive all floors are approached and their positions are detected. Additionally, a correction value for the position sensor (APS-offset) is calculated. The menu guides you after the start through the necessary steps
- For linear position sensors the distance between a concrete number of measured points has to be set for the parameter *APS-length* and for the parameter *APS-resolution* the number of measurement points that are set for *APS-length*
- Even blind floors must be approached during a learning journey

## In-/Outputs

Depending on the model, flexyPage displays provides inputs and outputs. These behave at the CAN bus like normal CANopen-Lift input/output boards. The associated function can be set in the submenu of the individual inputs/outputs as well as the associated floor, door and lift. To send a message to the car, floor "0" must be chosen and to send a message to all floors, choose "floor 255". In addition, the inputs / outputs can be inverted. In this case, they behave like conventional inputs / outputs. Changes to the terminals are then not sent to the CAN bus, but handled internally. Calls continue to respond to acknowledgments received via CAN bus.

## ELFIN Call Controller

If you want to connect the display via an in-/output module, and not via a CANopen compatible controller, some settings need to be carried out in the menu *ELFIN Call Controller*. This requires an additional CANopen-Lift in-/output board that needs to be connected with the controller. Additionally, the CAN interface needs to be connected with the flexyPage display. The board will translate the outputs from the controller in CAN messages and send them to the display via the CAN bus. This, in turn, can set the inputs of the controller in the opposite direction.

CANopen-Lift IO-nodes are the most common devices to connect input and output modules with the CANopen controller. The opposite, namely to use a CANopen IO node with a controller with IOs / without CANopen, is usually not possible. For this reason, a part of the car calls is used to transfer the IO states. The ELFIN Call Controller converts then this information into the appropriate function. Afterwards, it is send to the CAN bus so that the communication with all other CANopen compatible devices can be established. When using the Call Controller, the maximum count of available floors reduces to 32. Therefore, the following information must be set for each in-/output node:

- IO 1 = Standard internal call (acknowledgement) Floor 33, Door 1, Lift 1

- IO 2 = Standard internal call (acknowledgement) Floor 34, Door 1, Lift 1
- IO 3 = Standard internal call (acknowledgement) Floor 35, Door 1, Lift 1
- ...
- IO 127 = Standard internal call (acknowledgement) Floor 159, Door 1, Lift 1

It is possible to use several IO-nodes. With two nodes of 8 IOs each, the first node would receive the calls from floors 33-40 and the second, from floors 41-48. It is also possible to divide a call on two boards. This allows the call signal of the displays to be transmitted to the controller via a node, other than the acknowledgment of the control to the display.



**The following settings may only be activated on a flexyPage display on the bus as this will transmit everything to all other flexyPage displays and other CANopen-Lift-capable devices. Particularly, when importing and exporting settings, please pay attention!**

## CANopen master

If CANopen Master is activated in the display, the display will also take over the function of the CANopen master. This function can be activated with only one device, since two masters would interfere with one another. The connection to the control must be monitored by the display, which controls the CANopen input/output nodes, since this is the only connection. For this reason, the node ID 16 or 17 must be assigned to the input/output node. Node ID 3 is assigned to a connected CANopen position sensor. If none of these nodes are present or one of them is suddenly no longer there, the display automatically goes into the *out of order* state.

## Node-ID

The ELFIN Call Controller appears on the CAN bus as a separate node. Therefore, it requires its own node ID. We recommend to use as ID 1.

## General I/O parameters

### Binary/graycode basis 0:

In older systems it may occur that for the reduction of the needed inputs for the floor indication, "0" instead of "1" was used as a basis. This is not used anymore because the leveling can now be used as error diagnosis.

## External CANopen inputs and outputs

If these are activated, and the inputs/outputs are configured, the ELFIN Call Controller converts the states of the inputs/outputs into CANopen-Lift messages and sends them to their own display, as well as to all other devices on the CAN bus.

In addition to the functions that are defined in the CANopen standard, it supports a feature to transfer the floor status as a binary or gray code. If this is supported by the controller, fewer outputs will be required to transfer the floor status.

On the CAN bus the ELFIN Call Controller is visible as a separate node. The individual inputs/outputs can be configured as in a normal CANopen-Lift input/output board. They can be found, as defined by the standard, under the object directory entries 0x6100h and 0x6200h. Changes made with a CAN configuration tool (for example CANwizard) must be saved by clicking *save all non-volatile parameters* and they become effective after a *node reset*.

In the ELFIN Call Controller, the command *node reset* does not work reliably, so this step needs to be performed on the input/output panel (ELFIN fpc-01). As a result, both nodes are restarted. However, the non-volatile memories must be saved separately for each of the nodes.

## Generate messages from position values

If a CANopen-Lift position sensor is connected with the flexyPage display and a learning drive has been performed, CANopen messages can be generated from the position values. Thanks to this, less input and outputs are necessary and it is possible to enable functions that the original controller did not support.

The following messages can currently be generated:

- floor status
- arrival indication
- speech synthesis
- direction indication

These functions may only be activated, if no the CANopen-Lift controller is used. Besides, like all ELFIN Call Controller functions, it may only be activated on one display per lift.

# Example topology and node-ID assignment

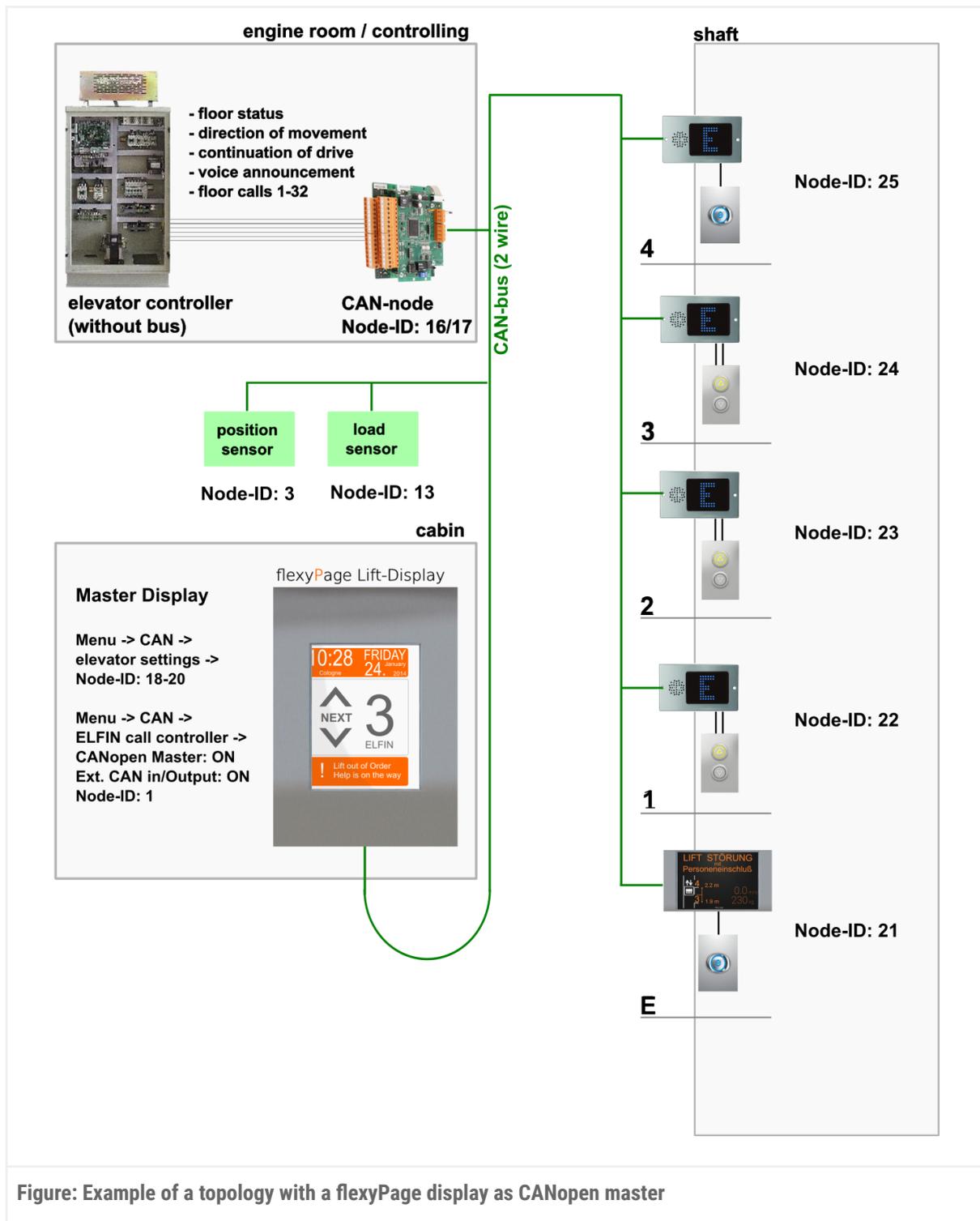


Figure: Example of a topology with a flexyPage display as CANopen master

# Energy saving functions

flexyPage displays provide various possibilities to reduce the energy consumption and extend the life span of the displays.

## **DVI/HDMI**

If a monitor is connected via HDMI, the flexyPage device can prompt the monitor via the HDMI cable, to switch into an energy-saving mode. This is supported by most monitors, but not by all.

## **LVDS**

The displays of the LT-Line, DM-Line and V-Line are controlled internally via LVDS interface. With this interface it is also possible to dim the brightness of the display. At 100% the display has the full brightness, at 0% it is completely switched off.

## CANopen-Lift

If the flexyPage displays are used in a system that supports the standard CANopen-Lift, the energy-saving functions should be controlled via the bus, through the control system. In the standard CANopen-Lift, eight different energy-saving modes are defined, of which the stages 1 (car display) and 4 (floor display) are relevant for the displays in most cases. If necessary, however, the other stages can also be used.

If no CANopen-capable controller is available, the display can automatically switch to a configurable energy-saving mode after a freely determinable time, if it did not receive any other status changes from the controller, such as, e.g. calls, special messages or floors. For each interface, an individual energy-saving stage can be configured, if used. Another alternative is the use of inputs (see discrete signals/inputs).

## Discrete signals / inputs

In addition to the CANopen-Lift variant, it is also possible to control the energy-saving function via discrete signals. Eight energy-saving levels can be controlled and whether the respective level is activated or deactivated via an input. In addition, the signals can also be used to set a wake-up from energy-saving mode.

The energy-saving signals, like other signals, can also be used to control the visibility of widgets on the display.

## Sound settings

All sound parameters can be set up under the menu item *Sound settings*. In addition, any number of playlists can be created. The starting point of the playback can be controlled via the timer system, that is also used with the widgets. Start and end times, data or days, as well as event control are possible. As a result, noise can be prevented by gongs or announcements at night.

Several sound files can be added to each playlist. The order of these, within the playlist, is indicated by the number in the upper corner of the sound file. In addition, a sound channel must be selected for each playlist. Currently, there are two:

- Standard signals
- Alerts / Emergency

Standard signals are played once, when the signal or event becomes active, as in the announcement of the floor level. When several signals or events are triggered simultaneously or in rapid succession, the playlists of the individual signals or events are automatically added to a global playlist and played one after the other, in the order in which they were triggered.

Alerts and emergency signals are repeated until the signal or event, triggered by the playlist is inactive. Standard signals are ignored or canceled.

A *background* sound canal will be available soon to play background music in the lift.

# Widgets

## General Information

### Images

The following formats can be displayed: png, jpg, svg, gif, bmp.

Animated GIFs can be displayed as well. However, it is not possible to display animated SVGs.

There are several ways to display images:

- *contain*

The image is adjusted so that it fits into the parent element, while maintaining the height-to-width ratio. Therefore, areas that are not covered by the image can occur at the edges.

- *cover*

The image will be adapted, considering the proportion to fill the parent element completely. Hence, some parts of the image can be cut off.

- *100% 100%*

The image will be scaled so that it completely fills the parent element. The height-to-width ratio can therefore change.

All image formats except GIF can be colored in order to match the font color. To do so, use the alpha mask of the image. An image without alpha values will be displayed as a colored square.

The maximum resolution of images may not exceed 1920x1080 pixels in landscape or 1080x1920 pixels in portrait mode.

Sound	<p>Currently, the following formats are supported:</p> <ul style="list-style-type: none"> <li>• .mp3 (recommended)</li> <li>• .wav (Can be used in most cases, but have to be tested individually, as there are differences between the various WAV formats and the flexyPage display generations. For devices of the type fpc-02, a 0.8-second long fade-out appears at the end of the sound file. This must be taken into account when creating the sound files)</li> </ul>
Colors	<p>Colors can be set as hexadecimal value in the format #AARRGGBB [Alpha,Red,Green,Blue] or in the color scheme selection menu. The right, vertical bar in the color selector can be used to adjust the transparency.</p>
Texts	<p>Texts adapt automatically to the size of the field in which they are written. If a text is too big for an area, use the set size to automatically reduce the size to fit into the element. Link breaks will automatically be inserted as well, to use the available space at the best.</p> <p>Additionally, it is possible to format the text with HTML/CSS commands.</p> <p>The list of theoretically supported commands can be found here:  <a href="http://doc.qt.io/qt-4.8/richtext-html-subset.html">http://doc.qt.io/qt-4.8/richtext-html-subset.html</a>(link is external)</p> <p>Alternatively, it is possible to configure the font size of the individual text widgets with pixel accuracy. This is particularly useful for static texts whose length does not change and where a specific text size is required due to a design specification or normative specifications.</p>
Fonts	<p>Some open source fonts are already preinstalled on the device. Additional fonts of the type .ttf and .otf can be installed by the user.</p>
Timer/Events	<p>It is possible to show or hide a widget in response to concrete time or events. Thereby, the widgets can be adjusted to date ranges (e.g. 01.03.2014-15.10.2016, a week or several days) or to concrete events (e.g. the lift is in floor 3 or the lift goes upwards).</p>

	<p>The parameters are combinable and it is possible to add several timers or events to a single widget as well.</p> <p>This makes the next situation possible, for example: A hotel has a terrace on the top floor which is opened every day but Sundays from 11:00 to 22:00 between April and October. Maintenance and cleaning services need access to the top floor outside of these hours.</p> <p>We set a timer for 11:00-22:00 for the terrace call button between April and October and choose every day but Sunday.</p> <p>For the maintenance and cleaning service we define a second timer/event, in this case the event SpecialFunction &gt; KeyLock 1. Now, the button will only be shown when a member of the maintenance or cleaning service identifies himself with a key.</p> <p>A widget for which no timer or events have been defined will always be displayed, except when there is a widget at a higher level that overlaps.</p> <p>Even when a widget is only partially being overlapped by another widget that is from a higher level, it will not be displayed.</p>
All widgets	<p>It is possible to choose a color and an image for each widget. The same applies to border, width and color. Additionally, padding can be also defined.</p> <p>Widgets can be docked to each other. This results in a compound belonging zone. The option to dock onto each other must be activated only in one of the widgets, otherwise they would overlap.</p> <p>The synchronisation of the allocation of a widget in a concrete lift or door in the WebEditor with the CANopen object dictionary does not work solidly for some widgets yet.</p> <p>In these circumstances, it must be adjusted in both, the widget and the CANwizard.</p>
All buttons and switches	<p>For each button or switch, besides the default image, a second picture can be defined for an acknowledged state. Also, border color, border width and background color can be defined for an acknowledged state.</p>

# Widget-Descriptions



## Lift-Buttons

### PIN Pad

Available on fpc-01, fpc-02, fpc-04, fpc-05 devices



For each button a PIN code can be applied. It may consist of 3 to 6 digits and can be set in the web editor in the submenu of every button.

When a PIN code is set, the flexyPage display will automatically show a pop-up window with a keyboard whenever someone clicks on that button. If the 3 to 6 digits are entered correctly, the call configured for this button will be performed. In case of three wrong tries the screen will be locked for 5 seconds. The colors of the pop-up will automatically be derived from the current color scheme.

When configuring the widget `SpecialFunctionButton[basic]`, the PIN code must be entered when turning on or off.

## Car Calls

### CarCall[v2]

Available on fpc-01, fpc-02, fpc-04, fpc-05 devices



The widget `CarCall[v2]` provides a button for interior or car calls of lifts. It portrays the tenants for the configured floor as entered into the house editor as well as the floor name. The tenants can be shown above or below the floor name. Moreover, the height of the area, portraying the floor name is adjustable.

The counterpart to the `CarCall[v2]` widget is the `CarCall [h2]` widget.

It fulfills the same function as `CarCall [v2]`, with the difference that tenants and floor description can be aligned vertically.

The following parameters can be set in the `CarCall[v2]` widget:

- Background color for tenant area acknowledged/not acknowledged
- Background color for floor area acknowledged/not acknowledged

- Border color acknowledged/not acknowledged
- Font (size, color) for tenant area acknowledged/not acknowledged
- Font (size, color) for floor area acknowledged/not acknowledged
- Background image for tenants and floor acknowledged/not acknowledged
- Background image size
- Height of floor number area (raster units)
- Arrangement of floor number area below name
- Padding
- Docking of widgets left, right, top or bottom
- Border width acknowledged/not acknowledged
- Floor to which this button is assigned
- Lift (1 to 8) in a group
- Call priorities: standard, low, high
- Doors (1, 2, 3, 4) for selective or opposite doors
- PIN Code
- PIN Code deactivatable on screen

## CarCall[h2]

Available on fpc-01, fpc-02, fpc-04, fpc-05 devices



The widget CarCall[h2] provides a lift button for an interior or car call for lifts. It portrays the tenants for the configured floor, as entered into the house editor, as well as the floor name. The tenants can be shown on the left or the right side of the floor name. Moreover, the width of the area portraying the floor name is adjustable.

The counterpart to the CarCall[h2] widget is the CarCall [v2] widget. It fulfills the same function as CarCall [h2], with the difference that tenants and floor description can be aligned horizontally.

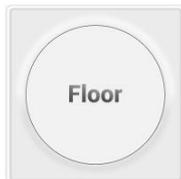
The following parameters can be set in the CarCall[h2] widget:

- Background color for tenant area acknowledged/not acknowledged
- Background color for floor area acknowledged/not acknowledged
- Border color acknowledged/not acknowledged
- Font (size, color) for tenant area acknowledged/not acknowledged

- Font (size, color) for floor area acknowledged/not acknowledged
- Background image for tenants and floor acknowledged/not acknowledged
- Background image size
- Width of floor number area (raster units)
- Arrangement of name left of floor area
- Padding
- Docking of widgets left, right, top or bottom
- Border width acknowledged/not acknowledged
- Floor to which this button is assigned
- Lift (1 to 8) in a group
- Call priorities: standard, low, high
- Doors (1, 2, 3, 4) for selective or opposite doors
- PIN Code
- PIN Code deactivatable on screen

## CarCall[floor]

Available on fpc-01, fpc-02, fpc-04, fpc-05 devices



The widget CarCall[floor] provides a button for interior or car calls of lifts. It shows the alias of the floor name as defined in the house editor.

The following parameters can be set in the CarCall[floor] widget:

- Border color acknowledged/not acknowledged
- Background color acknowledged/not acknowledged
- Font (size, color) acknowledged/not acknowledged
- Background image acknowledged/not acknowledged
- Background image size
- Padding
- Docking of widgets left, right, top or bottom
- Border width acknowledged/not acknowledged
- Floor to which this button is assigned
- Lift (1 to 8) in a group
- Call priorities: standard, low, high
- Doors (1, 2, 3, 4) for selective or opposite doors
- PIN Code

- PIN Code deactivatable on screen

### CarCall[tenant]

Available on fpc-01, fpc-02, fpc-04, fpc-05 devices



The widget CarCall[tenant] provides a button for interior or car calls of lifts. It displays the tenants for the configured floor as defined in the house editor as a text or a logo.

The following parameters can be set in the CarCall[tenant] widget:

- Border color acknowledged/not acknowledged
- Background color acknowledged/not acknowledged
- Font (size, color) acknowledged/not acknowledged
- Background image acknowledged/not acknowledged
- Background image size
- Padding
- Docking of widgets left, right, top or bottom
- Border width acknowledged/not acknowledged
- Floor to which this button is assigned
- Lift (1 to 8) in a group
- Call priorities: standard, low, high
- Doors (1, 2, 3, 4) for selective or opposite doors
- PIN Code
- PIN Code deactivatable on screen

### CarCall[basic]

Available on fpc-01, fpc-02, fpc-04, fpc-05 devices



The widget CarCall[basic] provides an interior call or car call of lifts. It can display a specific background image depending on the state *acknowledged* or *not acknowledged*. In addition, this widget can be used to output text regardless of the tenants that were set in the house editor.

The following parameters can be set in the CarCall[basic] widget:

- Border color acknowledged/not acknowledged

- Background color acknowledged/not acknowledged
- Font (size, color) acknowledged/not acknowledged
- Background image acknowledged/not acknowledged
- Background image size
- Padding
- Docking of widgets left, right, top or bottom
- Border width acknowledged/not acknowledged
- Floor to which this button is assigned
- Lift (1 to 8) in a group
- Call priorities: standard, low, high
- Doors (1, 2, 3, 4) for selective or opposite doors
- PIN Code
- PIN Code deactivatable on screen
- Text

## CarCall[Image]

Available on fpc-02, fpc-04, fpc-05 devices



The CarCall[Image] widget provides a call button that loads its background images from an URL.

The following parameters can be set in the CarCall[image] widget:

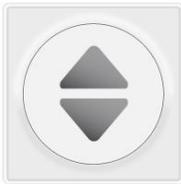
- Border color acknowledged/not acknowledged
- Background color acknowledged/not acknowledged
- Background image acknowledged/not acknowledged via URL
- Interval at which the images available at the URL is reloaded
- Background image size
- Padding
- Docking of widgets left, right, top or bottom
- Border width acknowledged/not acknowledged
- Floor to which this button is assigned
- Lift (1 to 8) in a group
- Call priorities: standard, low, high
- Doors (1, 2, 3, 4) for selective or opposite doors

- PIN Code
- PIN Code deactivatable on screen

## Hall Calls

### HallCall[basic]

Available on fpc-01, fpc-02, fpc-04, fpc-05 devices



The widget HallCall[basic] provides a button for a floor call or exterior call. It can issue direction-dependent calls (two interfaces: upwards and downwards) and direction-independent calls (one interface for calls). Type and size of the arrows can be freely configured. A freely configurable background image can be shown depending on the state acknowledged or not acknowledged.

The following parameters can be set in the HallCall[basic] widget:

- Border color acknowledged/not acknowledged
- Background color acknowledged/not acknowledged
- Background image acknowledged/not acknowledged
- Background image size
- Padding
- Docking of widgets left, right, top or bottom
- Border width acknowledged/not acknowledged
- Floor to which this button is assigned
- Lift (1 to 8) in a group
- Call priorities: standard, low, high
- Direction of the Hall Call
- Doors (1, 2, 3, 4) for selective or opposite doors
- PIN Code
- PIN Code deactivatable on screen

## Guest calls

### GuestCall[basic]

Available on fpc-01, fpc-02, fpc-04, fpc-05 devices



The GuestCall[**basic**] widget provides a call button for guest calls (including call / sender calls). The text displayed on the button can be customized. It can display a corresponding adjustable background image depending on the status acknowledged or not acknowledged. The pick-up floor and door and the target floor and door can be configured. The flashing which is common in some tenant calls states is generated by removing and adding the acknowledgment.

The following parameters can be set in the GuestCall[**basic**] widget:

- Border color acknowledged/not acknowledged
- Background color acknowledged/not acknowledged
- Font (size, color) acknowledged/not acknowledged
- Background image acknowledged/not acknowledged
- Background image size
- Padding
- Docking of widgets left, right, top or bottom
- Border width acknowledged/not acknowledged
- Source door (only in the Setup role)
- Destination floor (only in the Setup role)
- Destination door (1 to 4) (only in the Setup role)
- Button label (only in the Setup role)
- Floor to which this button is assigned (only in the Setup role)
- Associated lifts (only in the Setup role)
- PIN Code (only in the Setup role)
- PIN Code deactivatable on screen (only in the Setup role)

## Destination calls

**DestinationCallButton[floor]**

Available on fpc-01, fpc-02, fpc-04, fpc-05 devices



The DestinationCallButton[floor] widget provides a call button for destination calls. It displays the floor alias set in the house menu.

It requires a CANopen-Lift-capable destination call control.

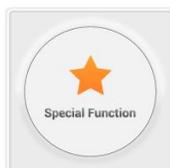
The following parameters can be set in the DestinationCallButton [floor] widget:

- Border color acknowledged/not acknowledged
- Background color acknowledged/not acknowledged
- Font (size, color) acknowledged/not acknowledged
- Background image acknowledged/not acknowledged
- Background image size
- Padding
- Docking of widgets left, right, top or bottom
- Border width acknowledged/not acknowledged
- Floor
- Lift (1 to 8) in a group
- PIN Code
- PIN Code deactivatable on screen

## Buttons + Switches

### SpecialFunctionButton[basic]

Available on fpc-01, fpc-02, fpc-04, fpc-05 devices



The widget SpecialFunctionButton[basic] provides a button/switch or indicator for various special functions. It can be configured, for example, for setting a lift as *out of order* or for longer periods of open doors for the loading of the lift.

There are three types of duration: active until operated anew (switch function), active for pre-determined period of time (timer function) and active only while being operated (button function).

The following parameters can be set in the SpecialFunctionButton[basic] widget:

- Border color acknowledged/not acknowledged
- Background color acknowledged/not acknowledged
- Font (size, color) acknowledged/not acknowledged
- Background image acknowledged/not acknowledged
- Background image size
- Padding

- Docking of widgets left, right, top or bottom
- Border width acknowledged/not acknowledged
- Floor
- PIN Code
- PIN Code deactivatable on screen
- Mode (Indicator only, selected while pressed, selected until pressed again, selected for certain time)
- Special function
- Text visible

### KeyPad for floor selection

**KeyPad[character]**

Available on fpc-01, fpc-02, fpc-04, fpc-05 devices



The KeyPadCharacterButton widget is used to enter characters for selecting floors. One or more characters can be configured for the button.

The following parameters can be set in the KeyPadCharacterButton widget:

- Border color acknowledged/not acknowledged
- Background color acknowledged/not acknowledged
- Font (color, size) acknowledged/not acknowledged
- Background image acknowledged/not acknowledged
- Wallpaper image size
- Inner distance (padding)
- Docking the widget left, right, up or down
- Margin width acknowledged/not acknowledged
- Text/characters for input
- Activate "Send without Enter button" mode - e.g. for buttons on the main floors

**KeyPad[enter]**

Available on fpc-01, fpc-02, fpc-04, fpc-05 devices



The KeyPadEnterButton widget is used to evaluate the characters previously entered with the KeyPadCharacterButton and to trigger the floor call.

The following parameters can be set in the KeyPadEnterButton widget:

- Border color acknowledged/not acknowledged
- Background color acknowledged/not acknowledged
- Font (color, size) acknowledged/not acknowledged
- Background image acknowledged/not acknowledged
- Wallpaper image size
- Inner distance (padding)
- Docking the widget left, right, up or down
- Margin width acknowledged/not acknowledged
- Button labeling
- InputOptions - The mapping of texts to be entered with signals which are to be triggered. Additionally, a pin code can be assigned for each text-signal combination

**KeyPad[clear]**

Available on fpc-01, fpc-02, fpc-04, fpc-05 devices



The KeyPadEnterButton widget is used to evaluate the characters previously entered with the KeyPadCharacterButton and to trigger the floor call.

In the widget KeyPadClearButton the following parameters can be set:

- Border color acknowledged/not acknowledged
- Background color acknowledged/not acknowledged
- Font (color, size) acknowledged/not acknowledged
- Background image acknowledged/not acknowledged
- Wallpaper image size
- Inner distance (padding)
- Docking the widget left, right, up or down
- Margin width acknowledged/not acknowledged
- Button labeling

**KeyPad[display]**

Available on fpc-01, fpc-02, fpc-04, fpc-05 devices



The widget KeyPadDisplay shows the previously entered characters and informs about the status of the floor selection process.

The following parameters can be set in the widget KeyPadDisplay:

- Border color
- Background color
- Font (color, size)
- Wallpaper
- Wallpaper image size
- Inner distance (padding)
- Docking the widget left, right, up or down
- Margin width acknowledged/not acknowledged
- Text - A freely configurable text as instructions for the user, e.g. "Please select floors and confirm with Enter".
- PinCodeText - A freely configurable text as instructions for the user, e.g. "Please enter pin and confirm with Enter".



## Lift-Status Indicator

### Direction Indicators

#### DirectionIndicator[double]

Available on fpc-01, fpc-02, fpc-04, fpc-05 devices



The widget DirectionIndicator[double] displays the direction of the run, the direction of movement and the direction indicator of a lift. If the lift is standing at a stop with a planned run in a specific direction, this is indicated by using an arrow and the message NEXT. If the car is moving in a specific direction it is indicated with the appropriate arrow without NEXT. When the lift is inactive, it is possible to show the arrows and text dimmed instead of hiding them.

#### DirectionIndicator[single]

Available on fpc-01, fpc-02, fpc-04, fpc-05 devices



The widget DirectionIndicator[single] portrays the direction of the lift's run using arrows. The design of the arrows for every direction can be freely configured.

#### EvacuationMovingDirection

Available on fpc-01, fpc-02, fpc-04, fpc-05 devices



The EvacuationMovingDirection widget displays the current direction of movement depending on the current change in position values. This can be used, for example, during an emergency rescue if the control signal is not available.

## DirectionIndicator[rotation]

Available on fpc-02, fpc-04



The DirectionIndicator[rotation] widget shows the direction of travel of the elevator car with an arrow graphic. When the direction of travel is changed, the transition is visualized by rotating the graphic. Due to the usage of a graphic the design of the arrow can be freely adjusted.

## Destination call indicators

### AcknowledgedFloors[horizontal list]

Available on fpc-01, fpc-02, fpc-04, fpc-05 devices



The widget AcknowledgedFloor[horizontal list] lists the floors on which the elevator will stop horizontally. The font size automatically adjusts to the number of floors to be approached. It is possible to determine a maximum number of floors to display, to prevent the font from becoming too small. It displays the floor alias set in the house menu.

### AssignedLift[text]

Available on fpc-01, fpc-02, fpc-04, fpc-05 devices



The AssignedLift[text] widget displays the name of the elevator that shall be used. The elevator designation or the text is freely configurable. It requires a CANopen-lift-capable destination call control.

## Announcements

### LiftInfo[icon+text]

Available on fpc-01, fpc-02, fpc-04, fpc-05 devices



The widget LiftInfo[icon+text] enables various announcements in a lift. It can show an image and a text at once. By default, the state of the doors will be displayed. Furthermore, all announcements described in the list can be displayed.

To activate an announcement for portrayal, it has to be activated in the menu *CAN > CANOpenSlave > MessageMapping*. With the occurrence of the specific signal, the determined announcement text and the image will be shown. If several events occur at once the portrayal has to be prioritized. At first the SpecialIndication with the highest priority will be shown. If there is no SpecialIndication active the next higher active SpecialFunction will be shown. If none of those is active the state of the doors is shown again. If the portrayal of a specific SpecialIndication or SpecialFunction is not desired, for example, the message *Occupied*, the text field of the message in the MessageMapping menu has to be emptied.

### LiftInfo[text]

Available on fpc-01, fpc-02, fpc-04, fpc-05 devices



The widget LiftInfo[text] enables the portrayal of various announcements of a lift. Similar to the widget LiftInfo[icon+text], it shows announcement texts of a lift, but without the possibility to output an image. The configuration and prioritization of the portrayal occurs as with the widget LiftInfo[icon+text].

### LiftInfo[icon]

Available on fpc-01, fpc-02, fpc-04, fpc-05 devices



The widget LiftInfo[icon] can be used to visualize various messages of a lift. It allows a flexible arrangement independent of the text to be displayed. The configuration and prioritization of the portrayal are done as with the widget LiftInfo [icon+text].

### SimpleMessageIndicator[text]

Available on fpc-01, fpc-02, fpc-04, fpc-05 devices



The SimpleMessageIndicator[text] widget displays a text received from the controller according to CANopen Lift standard.

## Floor Indicators

### FloorIndicator[alias]

Available on fpc-01, fpc-02, fpc-04, fpc-05 devices

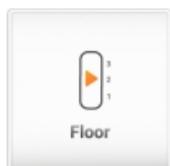


The widget FloorIndicator[alias] displays the current floor. The floor name is obtained from the field alias in the house editor. If a picture is configured for a floor, it will be displayed. If no picture or text is configured for the floor, the previous content will be displayed so that the user who drives through the floor does not notice if, for example, a floor is in maintenance. The layout can be configured in various ways, for example, font type and size, background color or image, frame color and much more are freely configurable.

The device types fpc-02 and fpc-04 the floor change can be animated. Under "Settings" you can enable this animation for a floor or car display. The scrolling movement for a floor display is shown in the direction of the movement of the elevator car. The scrolling movement in the car is animated against the direction of movement of the elevator.

### FloorIndicator[bar]

Available on fpc-01, fpc-02, fpc-04 devices



The widget FloorIndicator[bar] displays the current floor as a graphic that is shifted linearly animated in a bar on the display. With this widget it is possible to display, for example, a cabin in the shaft. The position of the animated graphics is independent of the actual position of the cabin and is based on the current floor level. An absolute position sensor is not necessary for this widget. The values for this widget are given in grid sizes to allow an accurate alignment with other widgets, such as call buttons for a particular floor. In order to be able

to display non-driveable building areas, an upper and a lower distance can be defined. To take account of blind floors, floors are skipped for which no alias has been entered in the house editor.

### FloorIndicator[nostalgic]

Available on fpc-01, fpc-02, fpc-04 devices



The widget FloorIndicator[nostalgic] displays the current floor as a nostalgic analogue indicator on the display. The layout can be configured in various ways, for instance font type and size, background color or image, hands, frame color and much more. The image with the pointer should always be as big as the dial with the floor descriptions. If the two images are superimposed, they will represent the current status of the lowest floor stand. Floors, for which no alias is entered in the house editor are skipped. The widget FloorIndicator[nostalgic] represents the current floor status animated, not the current position of the car in the shaft.

### SimpleFloorIndicator[text]

Available on fpc-01, fpc-02, fpc-04, fpc-05 devices



The SimpleFloorIndicator[text] widget displays a floor status sent as text by the controller according to the CANopen Lift standard.

## Lift-Status

### LiftStatus[classic]

Available on fpc-01, fpc-02, fpc-04, fpc-05 devices



The LiftStatus [classic] widget displays all the classic status indicators for a lift (floor, direction and messages). The design of the layout can be adjusted according to your own wishes. For example, font type and size, background color or image, frame color and much more are freely adjustable. The floor name can be edited freely in the house editor and thus be adjusted according to the requirements of the specific building. The desired special texts in the

announcement window can be chosen from a long list, as in the widget LiftInfo[icon+text].

## Info + Adverts

### FloorInfo[v2]

Available on fpc-01, fpc-02, fpc-04, fpc-05 devices



The widget FloorInfo[v2] displays an image and two texts depending on the current floor. The image extends along the whole background. One text is arranged at the upper area of the widget, the other one is arranged at the lower area. The height of the text areas is adjustable. Thus, the widget is ideally suited to portray floor-specific advertising or information.

The following parameters can be set in the FloorInfo[v2] widget:

- Floor (1 to 254)
- Layout (Padding, border width)
- Height of the text field in (can be adjusted using grid units)
- Border and background color (so as their transparency)
- Font type, font size, font color, direction and position
- Floor-dependent display of images and text, even with empty floors (e.g. for maintenance purposes) is the display of information is possible
- Floor-related test run allows you to preview of all previously set floor information as it will be shown on the display
- Timers and events

### FloorInfo[stream]

Available on fpc-02, fpc-04 devices



The FloorInfo[stream] widget displays a video network stream depending on the current floor or the target floor.

A stream source can be specified for each floor.

**Current floor" mode:**

If no source is specified for a "current floor", the previously played stream will continue to be displayed.

**Destination Floor" mode:**

When the display reaches the destination floor, the stream is displayed until the floor level changes or a new destination floor is found.

The following formats are supported:

- m3u8 playlists (h264) (fpc-02)
- rtmpt (h264) (fpc-02)
- rtsp (h264) (fpc-02 + fpc-04)

Due to the technology used, it can take a few seconds to set up and switch between the video streams.

Since there are also differences in the supported formats, please check each stream source after changing the configuration.

**Notes:**

Even if several widgets of this type and of the type "VideoPlayer[basic]" can be slid onto the surface, never more than one widget may be active at the same time.



## Entertainment

### Campaign[basic]

Available on fpc-02, fpc-04, fpc-05 devices



The Campaign[basic] widget enables the display of centrally managed media campaigns.

A permanent internet connection and an account for campaign.flexyPage.net are required to use this widget.

The following parameters can be set in the Campaign[basic] widget:

- Border color
- Background color

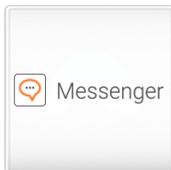
- Background image
- Background image size
- Inside spacing (padding)
- Widget docking left, right, top or bottom

For more information about flexyPage Campaign, please refer to flexyPage Campaign user manual.

## Web services

### Messenger[basic]

Available on fpc-02, fpc-04, fpc-05 devices



The flexyPage Messenger widget enables text-based communication with an emergency call center. The design of the widget is fixed. Only the size and position can be set freely.

To use this widget you need a permanent Internet connection and an account for messenger.flexyPage.net or a remote site that supports the Messenger protocol.

[Learn more about the flexyPage Messenger](#)

### Tenants[basic]

Available on fpc-02, fpc-04, fpc-05 devices



The Tenants[basic] widget enables the display of centrally managed tenant directories.

A permanent internet connection and an account for campaign.flexyPage.net are required to use this widget.

The following parameters can be set in the Tenant[basic] widget:

- Border color
- Background color
- Font (color, size)
- Background image size
- Inside spacing (padding)
- Docking widget left, right, top or bottom

For more information about flexyPage Campaign feature, please refer to flexyPage Campaign user manual

#### RssReader[header]

Available on fpc-01, fpc-02, fpc-04, fpc-05 devices



If an internet connection is available the widget RssReader[header] displays the *title* elements or headlines of an RSS feed. Besides, the source URL of the feed and the rate of updating the design of the widget can be adjusted according to your needs. For example, font type and size, background color or image, frame color and much more are adjustable. The given URL must always begin with "http://".

#### Weather[icon]

Available on fpc-01, fpc-02, fpc-04, fpc-05 devices



If an internet connection is available the widget Weather[icon] displays the weather conditions of a place for the current day or the next 5 days as an icon. This widget can be combined with the widget [Weather\[text\]](#) to flexibly realize various portrayals of weather information. For the widget Weather[icon] we provide you with various icon sets, which we will gladly adjust for your individual application.

#### Weather[text]

Available on fpc-01, fpc-02, fpc-04, fpc-05 devices



If an internet connection is available the widget Weather[text] displays the current temperature for a place or the minimum and maximum temperature for the next 5 days as digits. This widget can be combined with the widget Weather[icon] to flexibly realize various portrayals of weather information.

## Multimedia

#### TextArea[basic]

Available on fpc-01, fpc-02, fpc-04, fpc-05 devices



With the widget `TextArea[basic]` any text can be displayed. The design can be configured in many ways. Font and size as well as the colours of the text, the frame and the background can be freely adjusted. In contrast to the `"TextArea[locked]"` widget, the content of this widget can also be changed by users with the Content role.

### `DiaShow[basic]`

Available on fpc-01, fpc-02, fpc-04, fpc-05 devices



The widget `DiaShow[basic]` can display one or more images. If several images are selected they will be displayed one after another in a freely configurable interval. The transition between two images is animated. On fpc-01, fpc-02 and fpc-04 devices the duration of the animation can be configured as well. The presentation can be displayed in full screen or on a widget window.

### `VideoPlayer[basic]`

Available on fpc-01, fpc-02, fpc-04 devices



With a previously on the display saved video, the `VideoPlayer[basic]` widget plays it in an infinite loop. Aspect ratio and sound level can be set. We recommend using videos in format h264. Normally, those are in mp4, avi, or mov format. For soundtrack, mp3 is recommended.

#### **Notes:**

- If the video file does not contain a soundtrack, the volume in the video widget must be set to "0". Otherwise the screen will remain black.
- It is not possible to play a video with sound and simultaneous sounds, such as voice announcements or acoustic signals. If you want to show a video and use sounds, you must set the volume of the video to "0".
- For fel1X devices, it is not possible to use events and timers in combination with this widget.
- Even if multiple widgets of this type and of the type `"Video[Stream]"` are pushed to the interface, there should never be more than one widget active.

## RemoteViewer[image]

Available on fpc-02, fpc-04, fpc-05 devices



The RemoteViewer [image] widget can display one or more images from a network source. This can be a static image of a webcam or a specific file that you can access via a URL. As with a slide show, multiple images can be set and transitions can be specified.

## Video[Stream]

Available on fpc-02, fpc-04 devices



The Video[Stream] widget plays a video network stream. As a source, a network camera can be used for example.

The following formats are supported:

- m3u8 playlists (h264) (fpc-02)
- rtmpt (h264) (fpc-02)
- rtsp (h264) (fpc-02 + fpc-04)

As there are differences in the supported formats, every stream source must be checked.

### Notes:

- This widget is only available for fel2X devices
- Even if multiple widgets of this type and of the type "Video[Stream]" are pushed to the interface, there should never be more than one widget active.

## FloorInfo[video]

Available on fpc-02, fpc-04 devices



The FloorInfo[video] widget displays videos depending on the floor. One video can be configured for each floor.

If a travel signal is switched on, the video of the current floor is played. When the video is finished, the system checks whether the elevator is still running (travel signal is present). If this is no longer the case, a signal configurable by the user can be output (e.g. "door open"). If the travel signal is still present, the video of the next stop is played. If this video is also played to the end, a new check is made to see whether the travel signal is still present. If this is not the

case, the signal configured by the user is output. Otherwise, the system waits until the motion signal disappears and then outputs the signal configured by the user.

The signal configured by the user is output for 500 ms each.

It must be ensured that no new movement is started before the signal configured by the user has been triggered and that the signals Movement, Floor level, Next stop are available.



## Sensors

### Load

#### Load[digit]

Available on fpc-01, fpc-02, fpc-04, fpc-05 devices



The widget Load[digit] displays the current payload of the lift in kg. To display the measured values, the lift needs to dispose of a load measuring sensor that provides the display the information via CANopen-Lift.

The Tara value, which can be defined in the CAN menu, will always be subtracted from the shown value. If a load limit is defined in the CAN menu, the respective message will be shown when the value is reached.

#### Load[needle]

Available on fpc-01, fpc-02, fpc-04 devices



The widget Load[needle] displays the current payload of the lift as a tachometer. The background and the needle, as well as the range from minimum to maximum, are freely configurable. To display the measured values, the lift needs a load measuring sensor that provides the display the information via CANopen-Lift.

The Tara value, which can be defined in the CAN menu, will always be subtracted from the shown value. The defined overload value works as maximum value for the scale. The dial must be set with this value.

The pivotal point of the pointer from the bottom border of the image is configurable. The image of the dial and the pointer must have the same size. When placed on each other, the pointer should be at 0 kg.

### Load[icons]

Available on fpc-01, fpc-02, fpc-04 devices



The widget Load[icons] displays the current payload of the lift in up to 10 levels by means of freely definable icons. These icons can be persons or bars, for example. To display these icons, the lift needs a load measuring sensor that provides the display the information in kilograms via CANopen-Lift. The Tara value, which can be defined in the CAN menu, will always be subtracted from the shown value. This widget displays different images depending on the car load. The last image is displayed when the set overload value is reached.

### Load[bar]

Available on fpc-01, fpc-02, fpc-04 devices



The widget Load[bar] displays the current payload of the lift by means of a bar. The design of the portrayal can be freely defined, for example, from green (no load) to red (overload). To display this bar, the lift needs a load measuring sensor that provides the display the information via CANopen-Lift. The Tara value, which can be defined in the CAN menu, will always be taken-off from the displayed load. Another option is to set, whether the bar shall move inwards to outwards or vice versa. Additionally, it is possible to adjust the side of the widget, from which the bar shall appear.

## Speed

### Speed[digit]

Available on fpc-01, fpc-02, fpc-04, fpc-05 devices



The widget Speed[digit] displays the current speed of the lift's car. The portrayal occurs as a digit with the respective unit. Currently, a portrayal of m/s, km/h

and mph can be chosen. To display the speed, the lift needs a position sensor that transmits the position values via CANopen-Lift to the display.

## Position

### Position[shaft]

Available on fpc-01, fpc-02, fpc-04 devices



If the display is connected to an absolute position sensor, the widget Position[shaft] can display the current position of the car in the shaft graphically enhanced in a window or in fullscreen mode.

Especially in larger buildings or in television towers with fast-driving lifts, this widget offers an interesting indication for passengers. To display the position in the shaft the lift necessitates a position sensor that provides the position values via CANopen-Lift to the display. In addition, a learning journey must be carried out.

### Position[digit]

Available on fpc-01, fpc-02, fpc-04, fpc-05 devices



The widget Position[digit] displays the current position of the car in the shaft. The widget can display the absolute height in the shaft as well as the distance to the next floor. To display the measured values the lift necessitates an absolute position sensor that provides the display the values via CANopen-Lift.

### DistanceToFloor[digit]

Available on fpc-01, fpc-02, fpc-04, fpc-05 devices



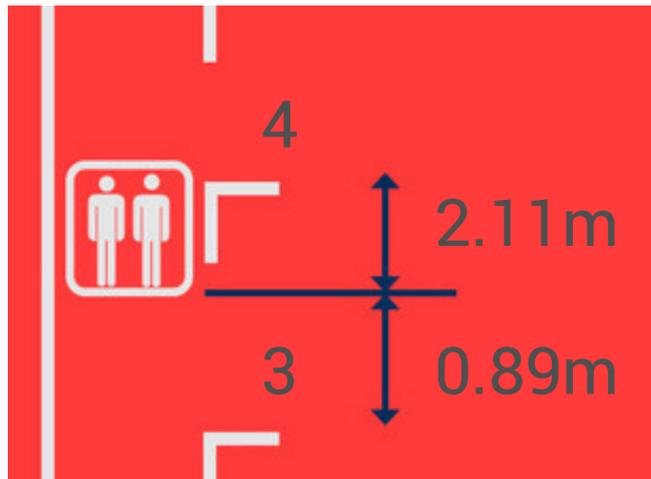
The DistanceToFloor [digit] widget shows the distance to a configurable floor. In order to display the distance, the elevator requires a position sensor, which provides the position values to the display via CANopen lift. In addition, a learning journey must be carried out.

## DistanceToAdjacentFloors[compact]

Available on fpc-01, fpc-02, fpc-04, fpc-05 devices



The widget DistanceToAdjacentFloors [compact] shows the distance between the elevator car and the next two floors. The areas destined for displaying the texts are fixed and not configurable.



**Figure: Picture of the widget DistanceToAdjacentFloors[compact]**

To be able to display the distance of the elevator car to the next floor shaft, the elevator requires a position sensor which provides the position values to the display via CANopen lift. In addition, a learning journey must be carried out.

## PositionImage[animated]

Available on fpc-01, fpc-02, fpc-04 devices



The widget PositionImage [animated] displays the frame of an animated GIF, depending on the current position of the elevator. Thereby, e.g. the ride of the cabin or general drive and position dependent animations can be simulated. For a good performance, the file size of the GIF should be as small as possible. The times between the frames defined in the GIF file are not taken into account. In order to display the appropriate frame for the position in the shaft, the elevator requires a position sensor which provides the position values to the display via CANopen lift. In addition, a learning journey must be carried out.

## Energy consumption

### EnergyConsumed[digit]

Available on fpc-01, fpc-02, fpc-04, fpc-05 devices



The widget EnergyConsumed[digit] represents the consumed energy as a number.

To display the measured values, the elevator requires an energy sensor which provides the display with the absolute consumption value via CANopen lift.

### EnergyFedBack[digit]

Available on fpc-01, fpc-02, fpc-04, fpc-05 devices



The widget EnergyFedBack[digit] displays the fed energy as a number. To display the measured values, the elevator requires an energy sensor that provides the absolute value to the display via CANopen lift.

### PowerSum[digit]

Available on fpc-01, fpc-02, fpc-04, fpc-05 devices



The widget PowerSum[digit] displays the current required energy as a number. To display the measured value, the elevator needs an energy sensor which provides the display with the absolute value via CANopen-Lift.

### PowerSum[bar]

Available on fpc-01, fpc-02, fpc-04 devices



The widget PowerSum[bar] displays the energy currently required in the form of a bar. The bar strikes from the center to the left when more energy is consumed than fed or from the center to the right when more energy is fed than consumed. The bar can be provided with an image or a specific color depending on the direction of deflection.

In order to display the measured value, the elevator requires an energy sensor that provides the display with the absolute value via CANopen lift.

## PowerSum[needle]

Available on fpc-01, fpc-02, fpc-04 devices



The widget PowerSum[needle] displays the currently needed energy as a pointer.

The pointer strikes from the center to the left when more energy is consumed than fed or from the center to the right when more energy is fed than consumed. The vertical pivot and the maximum deflection are configurable. The zero point is on the middle.

To be able to display the measured value, the elevator requires an energy sensor that provides the absolute values to the display via CANopen lift.



## Date + Time

## Date+Time[text]

Available on fpc-01, fpc-02, fpc-04, fpc-05 devices



The widget Date+Time[text] displays the current date. Its design and function can be configured in various ways. Besides the setting of font type and size, the color and background images, the date and time components can be combined with each other as well and they can be shown with any separator. For example: yy-MM-dd HH:mm:ss > 14-07-28 16:53:20.

### The following data are configurable:

Year	<ul style="list-style-type: none"><li>• yy → 14</li><li>• yyyy → 2014</li></ul>
Month	<ul style="list-style-type: none"><li>• M → 7 (without foremost null)</li><li>• MM → 07 (with foremost null)</li><li>• MMM → Jan. (localized shortcut of the month name)</li><li>• MMMM → January (localized month name)</li></ul>

Day	<ul style="list-style-type: none"> <li>• d → 8 (without foremost null)</li> <li>• dd → 08 (with foremost Null)</li> <li>• ddd → Mon. (localized shortcut of the weekday)</li> <li>• dddd → Monday (localized weekday)</li> </ul>
Hour	<ul style="list-style-type: none"> <li>• h → 2 (am/pm sensitive representation)</li> <li>• hh → 02 (am/pm sensitive representation with foremost null)</li> <li>• H → 8 (24 hours representation without foremost null)</li> <li>• HH → 16 (24 hours representation with foremost null)</li> </ul>
Minute	<ul style="list-style-type: none"> <li>• m → 3 (without foremost Null)</li> <li>• mm → 03 (with foremost Null)</li> </ul>
Second	<ul style="list-style-type: none"> <li>• s → 4 (without foremost null)</li> <li>• ss → 04 (with foremost null)</li> </ul>
AM/PM	<ul style="list-style-type: none"> <li>• a → <i>am / pm</i></li> <li>• A → <i>AM / PM</i></li> </ul>



**Note:**

For the am / pm sensitive representation *h* or *hh* must be used.

**Date+Time[stored]**

Available on fpc-01, fpc-02, fpc-04, fpc-05 devices



The Date+Time [stored] widget displays a permanently stored date. This date can be linked to an event (e.g.: "Maintenance mode"), and thus be updated. The display could then show, for example, "Last maintenance on dd.mm.yyyy". Additionally, the date can be offset (stored date + 12 months), so that the information "Next maintenance: mm.yyyy" can be displayed.

**Time[analogue]**

Available on fpc-01, fpc-02, fpc-04, fpc-05 devices



The widget Time[analogue] displays the current time as an analogue clock. The background and the hands can be configured individually. This makes it

possible to display the clock in classic or modern form, adapted to one's own needs.

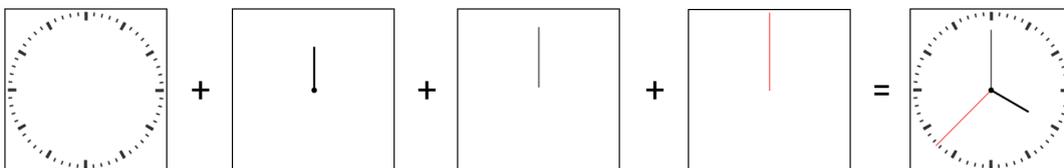
The following parameters can be set in the Time[analogue] widget:

- Layout (Padding, border width)
- Border and background color (as well as their transparency)
- Background image
- Dial, hour-, minute- and second hands (displayed at 12 o'clock) are uploaded in single, transparent images and displayed superimposed in the widget
- Timer und Events when the widget is displayed

### Sample application for the widget time [analogue]



1. Set background color and border color of the widget under Colors, images, font to transparent.
2. Select dial, hour hand, minute hand and second hand from the widgets library or upload them as single, transparent images (Attention: Hands must point to 12 o'clock!) and add to the widget.
3. If, for instance, a dial and three additional pictures for the individual pointers are selected, the result looks as follows:



## Miscellaneous

**TextArea[locked]**

Available on fpc-01, fpc-02, fpc-04, fpc-05 devices



The widget TextArea[locked] enables the portrayal of any text. The design can be configured in various ways. Font type and size, as well as color of the text, the frame and the background are freely adjustable. In contrast to the widget TextArea[basic] the text cannot be changed with the content role. This makes it particularly suitable for content that is only to be changed by the elevator mechanic.

### NextStopIndex

Available on fpc-01, fpc-02, fpc-04, fpc-05



The widget NextStopIndex shows the nth next stop in a controller-managed list of next stops. It can be used in combination with several of these widgets, for showing additional next stops. So the composition is flexible. In "Settings" the "Index in Next Stop List" setting an index can be set. This index will determine which position in the list should be displayed.

### JourneyCounter[digit]

Available on fpc-01, fpc-02, fpc-04, fpc-05 devices



The JourneyCounter[digit] widget can be used to display the journey counter transferred from the controller according to the CANopen standard. The font and size as well as the colours of the text, the frame and the background can be set freely.

### JourneyDurationCounter[digit]

Available on fpc-01, fpc-02, fpc-04, fpc-05 devices



The JourneyDurationCounter[digit] widget can be used to display the operating hours counter transferred from the controller in accordance with the CANopen standard. The font and size as well as the colours of the text, the frame and the background can be set freely.

## Games

### Tic Tac Toe

Available on fpc-01, fpc-02, fpc-04, fpc-05 devices

The widget Tic Tac Toe is the well-known game. You play against the computer.

The color of the boxes (background) and the font or line color can be set.

Transparent backgrounds and background images are not possible.

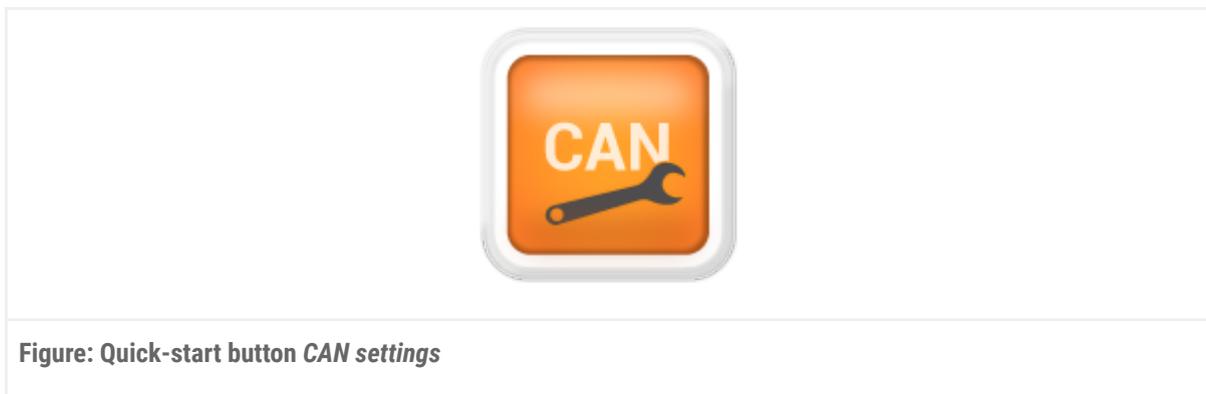
## Quick start guides

### Configuration of IOs

The following text describes how an input can be translated into a special message in the web configurator.

#### CAN configuration set up

The CAN configuration can be accessed via the quick-start button *CAN settings* (see figure).



The submenu *Lift Settings* is selected in the menu that opens.

#### IO mappings set up

The menu item *Invert IOs* needs to be activated. There are six IO slots available in the *IO Mapping* submenu. These slots define how to react to a signal from the respective slot. Open the *IO 1* tab by clicking on the grey area, marked *IO 1*. The following settings need to be adjusted (also see following figure):

- *Type* → *Input*
- *Basic function* → *Special indication*
- *Indication* → *user defined 1*
- *Associated lifts* → *1*
- *Floor* → *car*
- *Associated doors* → *1*

With these settings it is now configured that, if a signal is present at IO1 on the flexyPage display, the special signal *User 1* is provided on the bus.

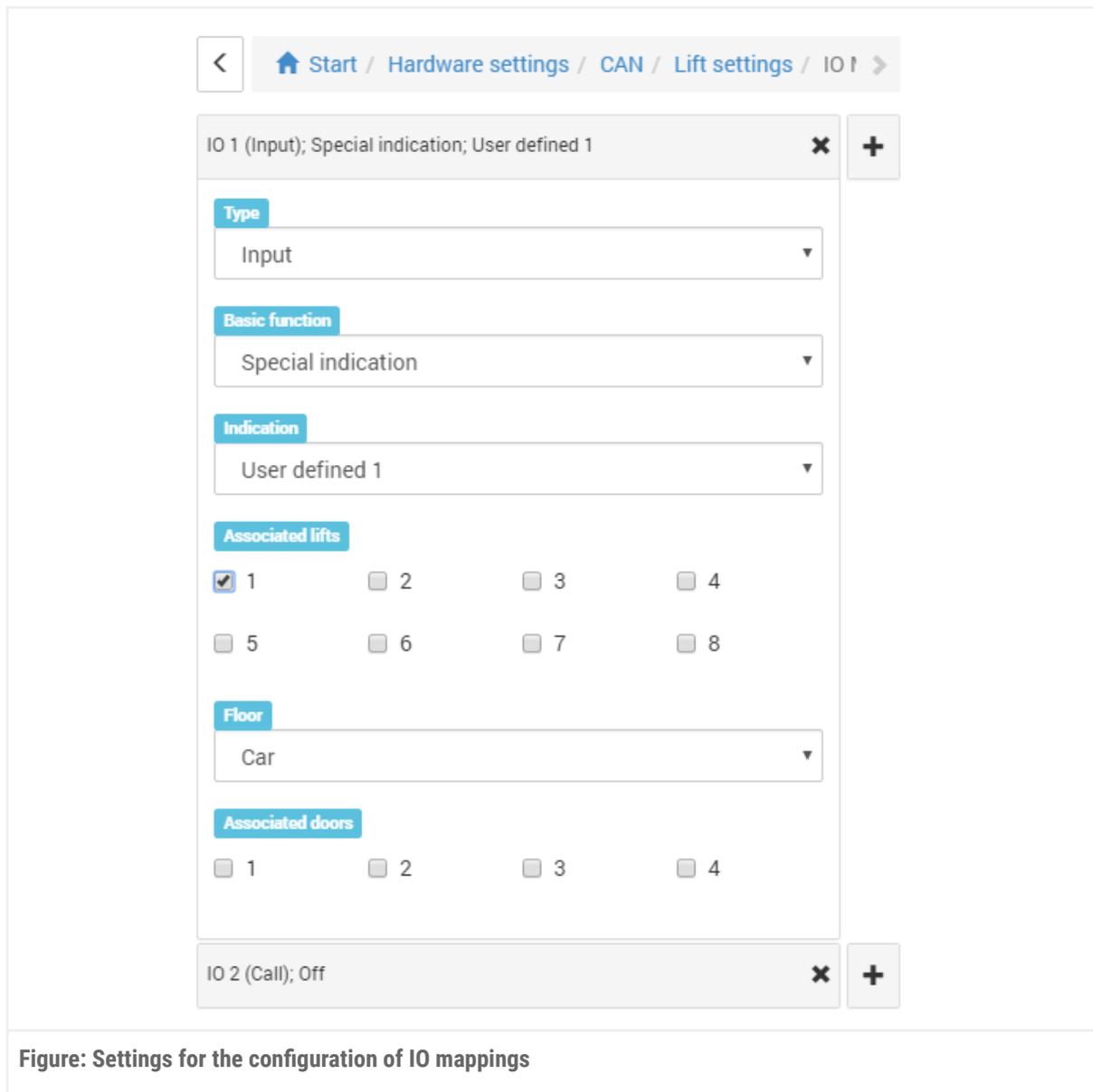


Figure: Settings for the configuration of IO mappings

## Show signal

To display this message on the display, one of the widgets *LiftInfo[icon+text]*, *LiftInfo[text]* or *LiftInfo[icon]* is required. This widget displays the message that is defined for *user-defined 1*.

## Configure signal

In the *Widgets and signal mapping* section, which can be accessed via the *widgets and signal mapping* button (see following figure), special signals can be set. To do this, open the *Signal mapping* menu and the *Special indications* menu. Scroll down to the *User defined 1* field and enter the text that shall be displayed on IO 1 when a signal is applied (see figure after next).

Optionally, an image can be set additionally or alternatively, which is also displayed in the *LiftInfo[icon+text]* or *LiftInfo[icon]* widgets.



Figure: Quick-start button *widgets and signal mapping*

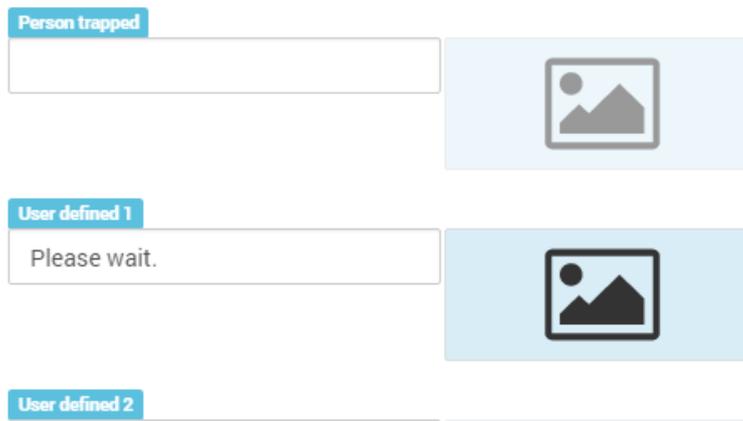


Figure: Preset text for signal *User defined 1*

## Common errors and solutions

### Lift & CAN



**Error:**

**The display does not respond to lift data or displays the message Out of Order.**

**1. The CAN bus is not working.**

- Check whether the CAN bus is correctly terminated.
- Check whether for all equipment has been set the corresponding baud rate.

## 2. The display cannot find a master.

- Check if the master is active. For this purpose, either there must be a CANopen controller or the entry *CANopen Master* in the menu *CAN > ELFIN Call Controller* must be active. The node IDs in the master must be between 1 and 8.

## 3. The inputs / outputs are incorrectly or not configured (only relevant with the use of additional input / output boards).

- Check I/O modules in the menu *CAN > ELFIN call controller* and the I/O modules of the external CANopen lift board, which is connected to controlling.

## 4. The node number is incorrect.

- Check in the CAN menu whether all nodes are set up correctly.
- Check other nodes on double nodes ID.
- With the use of additional input/output modules: check if the CANopen input/output boards have the nodes 16 and 17 and the position sensor (if any) have the node ID 3.



### **Error:**

**The display shows the wrong lift status.**

## 1. The wrong lift is set (only with lift groups).

- Check the lift number in the menu *CAN > lift settings > assigned lifts*.

## 2. The wrong floor is set.

- Check which floor is set in the menu *CAN > lift settings > floor*.
- The car is floor 0, the lowest floor is 1, ... .

### 3. The inputs/outputs are incorrectly configured.

- Check if the function on each input/output is correct. It could happen that the functions were misplaced in the configuration.

### 4. CANopen master / External CAN inputs / outputs are activated for more than one display.

- Disable the CANopen master function for all displays except one. If several CANopen Master are turned on simultaneously, these will interfere with each other.

## Display



### Error:

**The floor is depicted wrong or not at all with the widget *FloorIndicator[alias]*.**

### 1. The floor is not configured.

- Check whether in the menu *house editor* all floors are set. Only those floors will be displayed, which have an alias. If an image is added, this will be displayed too. If neither image, nor text have been entered, the previous text will be displayed so the user doesn't realize for example that he passed a maintenance floor.

### 2. No floor position is provided or the controlling sent a floor level = 0.

- floor level = 0 means nothing should be displayed. Hence, the behaviour is correct. At operation with in- and output it can happen that a different floor level is being sent. In this case in- and output on the external CAN In- and output board as well as in the menu *CAN > ELFIN Call Controller* need to be checked. Attention should especially be paid to the chosen format of the format of the floor level (standard, Graycode, binary).



**Error:**

**No floor information is depicted with the widget FloorInfo[v2].**

**1. No information for the floor has been provided.**

- Check whether the floor information has been provided in the widget FloorInfo[v2]. If nothing has been set for one floor, nothing will be displayed.

## Glossary

### Widget

A **software widget** is a relatively simple and easy-to-use software application or component made for one or more different software platforms.

A desk accessory or applet is an example of a simple, stand-alone user interface, in contrast with a more complex application such as a spreadsheet or word processor. These widgets are typical examples of transient and auxiliary applications that don't monopolize the user's attention. On the other hand, graphical control elements (GUI *widgets*) are examples of reusable modular components that are used together to build a more complex application, allowing programmers to build user interfaces by combining simple, smaller parts.

*Source: de.Wikipedia.org*

# Your contact persons

Even an extensive documentation cannot answer all questions. Do you have questions or suggestions concerning our flexyPage system? We look forward to your requests. You can contact us at:

## **ELFIN Technology GmbH**

Im Zollhafen 22

50678 Cologne

Germany

Phone: +49 (221) 6778932-0

FAX: +49 (221) 6778932-2

service@elfin.de

www.elfin.de



## **flexyPage Sales**

flexypage.de/en/sales

Tel.: +49 (221) 6430816-2

FAX: +49 (221) 6778932-2

sales@flexyPage.de



innovative display solutions

## **flexyPage Support**

flexypage.de/en/support

Phone: +49 (221) 6430816-3

support@flexyPage.de