

# README - P44-LC-X

## Context

[plan44.ch](https://plan44.ch) produces *and lives from selling* ready-to-use digitalSTROM gateway products, the P44-DSB series, and standalone light and automation controllers (P44-LC and P44-AC series), focusing on DALI, EnOcean, and hue components.

**Starting with beta versions 1.7.0.x, P44-LC devices also have support for the new matter home automation standard, see <https://plan44.ch/matter>.**

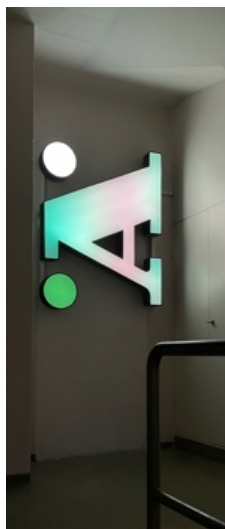
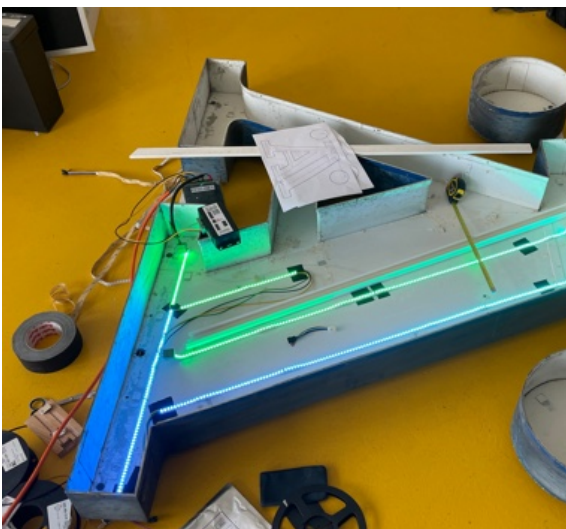
However, custom P44-LC and P44-AC controllers are often used with addressable SmartLED chains (WS28xx) for more advanced lighting and automation installation. To see examples, have a look at <https://plan44.ch/custom>.

While current P44 products are custom hardware, all are based on OpenWrt which also runs on RaspberryPi (in fact, the now legacy P44-DSB-E was itself based on RaspberryPi B+).

For many years already, the free [P44-DSB-X](#) image for RaspberryPi allows DIY users to extend their digitalSTROM systems with a lot of custom hardware with very little effort. Any GPIO can be made a input or button with a few clicks, SmartLED chains become full-featured light devices with a nice web interface, and the built-in p44script language allows almost any kind of integration of http APIs, DC-motors, i2c and SPI devices etc.

**As of beta version 1.7.0.39, the P44-LC devices support MIDI (via USB or MIDI RTP via network), see [https://plan44.ch/p44-techdocs/de/script\\_ref/#midi](https://plan44.ch/p44-techdocs/de/script_ref/#midi).** This allows combining light and sound effects, such as controlling lights from a DAW timeline or triggering sounds from events in the P44-LC. Also, MIDI controllers with analogue faders and knobs are very useful input devices for tuning all sorts of light effects - these can be integrated into P44-LC setups with a few lines of p44script code.

Now, with the P44-LC-X image, all these features, including the high-level abstraction of "zones", "scenes" and "triggers" can also be used in a completely standalone way (as with the P44-LC products). Just flash an SD card with the P44-LC-X image, boot it up, and you have a small home automation system, as well as a platform for integrating many types of devices with each other to build more complex setups.



an old signage "Ä", visualizing weather with color LEDs, controlled by a P44-LC

The controller functionality is implemented in the plan44 **vdcd**, a GPLv3 open source (<https://plan44.ch/opensource/vdcd>) implementation for digitalSTROM interfacing. For becoming a standalone controller, the **--localcontroller** command line option enables local "zone" and "scene" management, so vdc becomes a small but full-featured home automation solution.

## vdcd

Of course, the vdc project needs to contain driver implementations for DALI, EnOcean and hue as supported by the P44-DSB and P44-LC products.

However, vdc has to offer a lot more, in particular for people willing to experiment and build their own stuff:

- **button devices** based on digital inputs such as **GPIO** or **i2c port extender chip pins** with built-in detection of single, double, triple, quadruple-clicks to directly call scenes, as well as click-and-hold for dimming, not only brightness, but any device channel, such as color.
- fully functional **dimmed lights** based on **i2c connected PWM chips** or **DMX512** devices via OLA (<http://www.openlighting.org/>, using an external USB-to-DMX adapter, e.g. DMXKing).
- fully functional **color lights** (RGB, RGBW or even RGBWA) based on i2c PWM or DMX512, or even more fancy "lamps" like the MessageTorch (<https://github.com/plan44/messagetorch>)
- **Smart LED chains** (WS28xx compatible chips) can be connected directly (Data for WS28xx chain is RPi P1 connector pin 12 = PWM = GPIO18) and then arranged to cover areas in many ways (wound to a cylinder, zigzagged to cover an area etc.). To see what's possible with the **p44lrgraphics** subsystem that powers color LED chains, see this youtube video: <https://youtu.be/vLZRrn7Mh5I> and have a look at the plan44 techdocs at <https://plan44.ch/p44-techdocs>
- **switching outputs** (e.g. for relays) based on **GPIO** or **i2c port extenders**
- **scripted devices** using the built-in p44script language to integrate devices and services e.g. via http REST APIs, websockets, , MIDI, serial ports (UART), UDP packets, i2c, SPI, modbus and even a number of highly exotic special purpose "features" (drivers for things like Swiss Railway splitflap displays, RFID readers). See the plan44 techdocs at <https://plan44.ch/p44-techdocs>.
- **external device API** to create devices as external scripts or programs which connect via a socket to the vdc. See the plan44 techdocs at <https://plan44.ch/p44-techdocs>.
- All these devices can be **directly controlled** via the Web-UI, **grouped** into "zones", **orchestrated** via "scenes" and **interconnected** via "triggers" that can respond to input signals, sensor values, various other events and also date and time (including time relative to sunrise/sunset).

## A P44-LC for Makers

So, besides being the base for plan44 commercial products, a RaspberryPi running a vdc is also a great starting point for building your own fancy lighting solution. I really want to encourage this, so that's what the P44-LC-X is about (the same way as the P44-DSB-X allows DIY for digitalSTROM).

- You just need to buy a Raspberry, a SD card, and copy the firmware image to a SD card!
- **No official support.** Of course, feedback by email and true questions are welcome - but not those of the "I'm too lazy, make my stuff work for me, fast!" type ;-)
- No one-click online updates guaranteed (there will be some, from time to time, but there's no entitlement to get them all).

- Version for [Onion Omega2](#) (that's what is inside the P44-LC) is planned.
- Beta versions with matter support available for download from <https://plan44.ch/automation/p44-lc-x.php>

## How to use

- Flash the **correct** image (the RPi B+ image does not run on Rpi 2/3 and vice versa!) onto a SD card of at least 1GB size. A good tool for flashing SD cards on all platforms is Etcher (<https://etcher.io/>)
- Insert SD card into a RaspberryPi, Model B, B+, 2+ or 3+ (**RPi 4 is not yet supported by OpenWrt at the time of writing**)
- connect to your LAN (**must have DHCP**)
- connect power **and wait ~1 min** (if you have connected a HDMI screen, you can watch the boot process). **Note:** after very first boot, reboot the device once again, because some configuration becomes active only then.
- If you are on macOS, use a tool like (plan44's) [LocalSites](#) app, or another DNS-SD browser to conveniently discover the P44-LC-X web interface. On Linux, use **avahi-browse**. On Windows, the P44-LC-X should show up in the **network neighbourhood** via uPnP-SD. And of course, with access to your router you can find the P44-DSB-X like any other **DHCP client device**.
- The **default login** for the P44-LC-X webinterface is p44ladmin (both user and pw)
- **You can login via ssh as root with password *experiment*** - you should change this when your RPi is accessible from the open internet! The ssh service is also advertises via DNS-SD, so on macOS you can see it in the "New Remote Connection..." window.
- You can access the console when you plug in a HDMI screen and connect a USB keyboard.
- On RPi3, you can enable WiFi client or access point by clicking "Network Settings..." with Cmd+Shift (macOS) / Ctrl+Shift (Windows) keys held down.
- For basic information about the operating concept (zones, scenes, devices, triggers), have a look at the P44-LC product manuals, available from <https://plan44.ch/automation/p44-lc-de.php>. Your RPi-based P44-LC-X does not have DALI or EnOcean (so you can skip these paragraphs), but it does have hue support and everything else described in that manual.
- For technical information, in particular regarding the p44script language, visit <https://plan44.ch/p44-techdocs>. There will be tutorials for simple and advanced P44-LC-X based projects, and there is already (at the time of writing this, December 2021) a lot of reference information and examples in English and German.
- The OS on the P44-DSB-X is a customized OpenWrt with little packages installed. So you can use it more or less like a regular OpenWrt on the command line. The default feed URLs point to the plan44 server providing the packages specific for the P44-LC, but you can add feeds like on any other OpenWrt box. See <https://openwrt.org> for general information about the OpenWrt embedded Linux distribution.
- The OLA (for DMX512) website is available at port 9090
- The vdc drives the DMX universe number 42, so to use DMX, you need to create universe 42 in OLA connected to your DMX512 output device.

## How to participate

- Use it! This already makes you part of the P44 device community :-)

- Ask questions, present your projects in the plan44 community forum <https://forum.plan44.ch/>
- Improve it - vdc4 is Open Source on github <https://github.com/plan44/vdc4>, as well as the plan44 OpenWrt feed at <https://github.com/plan44/plan44-feed>.
- Provide feedback ([automation@plan44.ch](mailto:automation@plan44.ch) email, pull requests in github, [@plan44ch](https://twitter.com/plan44ch) on twitter)