

Ausgangslage

- es gab den Bedarf mehrere Projektoren synchron zu schuttern
- es sollte in allen Räumen gleich funktionieren
- der Einrichtungsaufwand bei Produktionen soll gering sein
- Ansteuerung über DMX mit dem grandMA3 Stellwerk

Suche nach Lösungen

- Nutzung der RS232 Schnittstelle der Projektoren
Die Kosten für Hardware die DMX512 zu frei programmierbaren RS232 Befehlen wandelt liegt bei knapp über 1000,- € pro Projektor
- Bei der Internetrecherche stießen wir auf den Artikel von Johannes Felber

Johannes benutzt einen Raspberry Pi in Kombination mit einem RS-232-TTL-Adapter
Und der Software des Open Lighting Architecture Projektes

Unsere Lösungen

- Wir entschieden uns für die Lösung von Johannes in abgewandelter Form
- Nutzung der in den Projektoren integrierten Webservern um den hohen Aufwand der RS232 Verkabelung bei Produktionen zu umgehen



Hardware

- Raspberry Pi 4 Model B
- Verkabelung über das vorhandene Lichtnetzwerk

Software

- Raspberry Pi Imager zur installation des Betriebssystems
- Open Lighting Architecture (OLA) aus den Paketquellen des Betriebssystems
- ssh für den Remotelogin
- Texteditor zum erstellen der Skripte und Config-Dateien
- shell zum Ausführen der Skripte
- curl (is a tool for transferring data from or to a server) Webserver des Projektors ansprechen
- netcat (arbitrary TCP and UDP connections and listens) Serial over Lan
- cron (Daemon zur geplanten Ausführung von Befehlen) starten von olad

Open Lighting Architecture

- erster Commit auf Github 2005
- läuft auf Linux, macOS und FreeBSD
- Konfiguration über Weboberfläche
- unterstützt viele Protokolle: ArtNet 1, 2 & 3, E1.31 (sACN), ESP Net, KiNET, Open Pixel Control, Open Sound Control, Pathport, SandNet, Strand Shownet
- Unterstützung für diverse USB-DMX512 Interfaces
- diverse Command-Line Tools wie z.B. ola_dmxconsole und ola_dmxmonitor nützlich bei der Entwicklung des eigenen Setups
- [OLA DMX Trigger](#) sehr nützlich zum Starten und Steuern anderer Programme
- API für C++ und Python



Weboberfläche ola



- Home
- Universes
 - 30
 - 45
- Plugins
 - ArtNet
 - Dummy
 - E1.31 (sACN)
 - ESP Net
 - Enttec Open DMX
 - FTDI USB DMX
 - GPIO
 - KarateLight
 - KiNET
 - Milford Instrumer
 - OSC
 - Open Pixel Contr
 - Pathport
 - Renard
 - SPI
 - SandNet
 - Serial USB
 - ShowNet
 - StageProfi
 - UART native DMX
 - USB

New Universe Settings

Universe Id

Universe Name

Available Ports

Device	Direction	Description
<input type="checkbox"/> Dummy Device	Output	Dummy Port
<input type="checkbox"/> ArtNet [192.168.50.112]	Input	
<input type="checkbox"/> ArtNet [192.168.50.112]	Output	
<input type="checkbox"/> ShowNet [192.168.50.112]	Input	ShowNet 1-512
<input type="checkbox"/> ShowNet [192.168.50.112]	Input	ShowNet 513-1024
<input type="checkbox"/> ShowNet [192.168.50.112]	Input	ShowNet 1025-1536
<input type="checkbox"/> ShowNet [192.168.50.112]	Input	ShowNet 1537-2048
<input type="checkbox"/> ShowNet [192.168.50.112]	Input	ShowNet 2049-2560
<input type="checkbox"/> ShowNet [192.168.50.112]	Input	ShowNet 2561-3072
<input type="checkbox"/> ShowNet [192.168.50.112]	Input	ShowNet 3073-3584
<input type="checkbox"/> ShowNet [192.168.50.112]	Input	ShowNet 3585-4096
<input type="checkbox"/> ShowNet [192.168.50.112]	Output	ShowNet 1-512
<input type="checkbox"/> ShowNet [192.168.50.112]	Output	ShowNet 513-1024
<input type="checkbox"/> ShowNet [192.168.50.112]	Output	ShowNet 1025-1536
<input type="checkbox"/> ShowNet [192.168.50.112]	Output	ShowNet 1537-2048
<input type="checkbox"/> ShowNet [192.168.50.112]	Output	ShowNet 2049-2560
<input type="checkbox"/> ShowNet [192.168.50.112]	Output	ShowNet 2561-3072
<input type="checkbox"/> ShowNet [192.168.50.112]	Output	ShowNet 3073-3584
<input type="checkbox"/> ShowNet [192.168.50.112]	Output	ShowNet 3585-4096
<input type="checkbox"/> ESP Net [192.168.50.112]	Input	
<input type="checkbox"/> ESP Net [192.168.50.112]	Output	
<input type="checkbox"/> SandNet [192.168.50.112]	Input	
<input type="checkbox"/> SandNet [192.168.50.112]	Output	
<input type="checkbox"/> Pathport [192.168.50.112]	Input	
<input type="checkbox"/> Pathport [192.168.50.112]	Output	
<input type="checkbox"/> E1.31 (DMX over ACN) [192.168.50.112]	Input	
<input type="checkbox"/> E1.31 (DMX over ACN) [192.168.50.112]	Output	
<input type="checkbox"/> OSC Device	Input	/dmx/universe/%d
<input type="checkbox"/> OSC Device	Input	/dmx/universe/%d
<input type="checkbox"/> OSC Device	Input	/dmx/universe/%d
<input type="checkbox"/> OSC Device	Input	/dmx/universe/%d
<input type="checkbox"/> OSC Device	Input	/dmx/universe/%d

Cancel Add Universe

Probleme

- bei gleichzeitigem shuttern von 4 Projektoren macht einer nicht mit (willkürlich)
Lösung: für jeden Projektor eine eigene Instanz von ola_trigger
- Latenz bei den Christie Projektoren (Webserver hat eine zu lange Reaktionszeit)
Lösung: Nutzen von Serial over Lan welches die Christie Projektoren implementiert haben

Auszug Christie Bedienungsanleitung



Serial API commands

SHU-Shutter

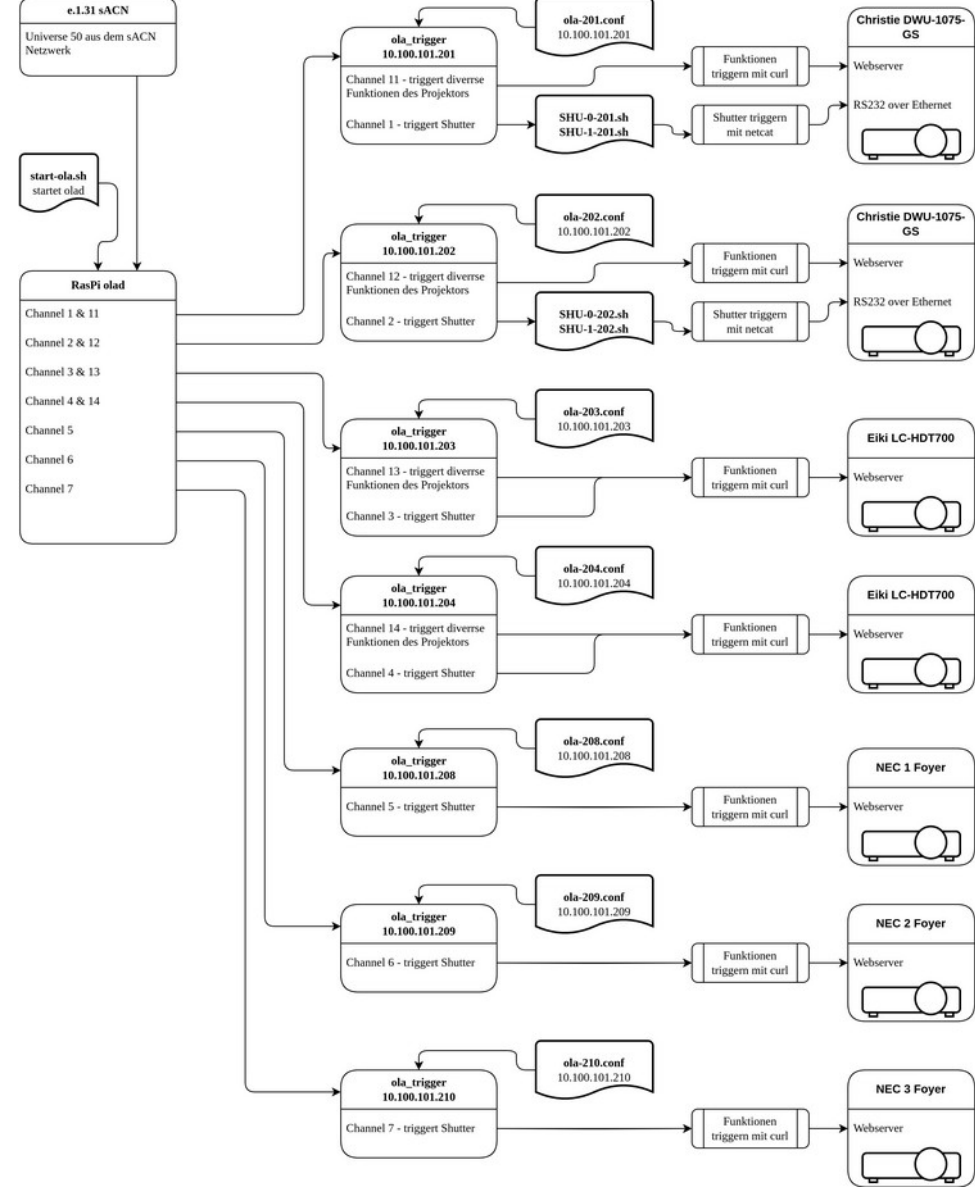
Opens and closes the shutter.

Commands

Command	Description	Values
SHU?	Gets the state of the shutter. (Read-only)	—
SHU <0 1>	Opens or closes the shutter.	0 = Opens the shutter 1 = Closes the shutter (Default)



Die Struktur der Lösung im FFT



Starten des ola Dämon mit cron bei Systemstart

```
# Edit this file to introduce tasks to be run by cron.  
#  
#  
# For more information see the manual  
#pages of crontab(5) and cron(8)  
#  
# m h dom mon dow  command  
@reboot /home/fft/start-ola.sh  
@reboot /home/fft/start-ola_trigger-201.sh  
@reboot /home/fft/start-ola_trigger-202.sh  
@reboot /home/fft/start-ola_trigger-203.sh  
@reboot /home/fft/start-ola_trigger-204.sh  
@reboot /home/fft/start-ola_trigger-208.sh  
@reboot /home/fft/start-ola_trigger-209.sh  
@reboot /home/fft/start-ola_trigger-210.sh
```

Die benötigten Dateien zum Betrieb unserer sieben Projektoren

```
start-ola.sh  
ola-201.conf  
ola-202.conf  
ola-203.conf  
ola-204.conf  
ola-208.conf  
ola-209.conf  
ola-210.conf  
SHU-0-201.sh  
SHU-0-202.sh  
SHU-1-201.sh  
SHU-1-202.sh  
start-ola_trigger-201.sh  
start-ola_trigger-202.sh  
start-ola_trigger-203.sh  
start-ola_trigger-204.sh  
start-ola_trigger-208.sh  
start-ola_trigger-209.sh  
start-ola_trigger-210.sh
```



start-ola.sh

```
#!/bin/bash  
sleep 40s && sudo killall olad && sleep 2s && olad
```

start-ola_trigger-201.sh

```
#!/bin/bash  
sleep 80s && ola_trigger -u 50 ola-201.conf &
```

ola-201.conf

11	5-12	`curl 'http://10.100.101.201/cgi-bin/webctrl.cgi.elf?&t:26,c:5,p:0,v:3`
11	13-25	`curl 'http://10.100.101.201/cgi-bin/webctrl.cgi.elf?&t:26,c:5,p:0,v:4`
11	26-38	`curl 'http://10.100.101.201/cgi-bin/webctrl.cgi.elf?&t:26,c:5,p:0,v:7`
11	39-51	`curl 'http://10.100.101.201/cgi-bin/webctrl.cgi.elf?&t:26,c:5,p:6,v:1`
11	52-64	`curl 'http://10.100.101.201/cgi-bin/webctrl.cgi.elf?&t:26,c:5,p:6,v:5`
11	65-76	`curl 'http://10.100.101.201/cgi-bin/webctrl.cgi.elf?&t:26,c:5,p:6,v:0`
11	77-89	`curl 'http://10.100.101.201/cgi-bin/webctrl.cgi.elf?&t:26,c:5,p:327715,v:1`
11	90-102	`curl 'http://10.100.101.201/cgi-bin/webctrl.cgi.elf?&t:26,c:5,p:458759,v:1`
11	154-166	`curl 'http://10.100.101.201/cgi-bin/webctrl.cgi.elf?&t:26,c:5,p:7,v:1`
11	167-179	`curl 'http://10.100.101.201/cgi-bin/webctrl.cgi.elf?&t:26,c:5,p:7,v:0`
1	128-255	`sh SHU-1-201.sh`
1	0-127	`sh SHU-0-201.sh`

SHU-1-201.sh

```
#!/bin/bash  
echo '(SHU 1)' | nc 10.100.101.201 3002 -N
```

SHU-0-201.sh

```
#!/bin/bash  
echo '(SHU 0)' | nc 10.100.101.201 3002 -N
```



Auslesen der Webserver Befehle mit den Web-Entwickler Tools in Firefox

CHKISTIE®

Model Name -
Serial Number -

Logout

user

Main

Network Setting

Tools

Administrator

About

General

Status

Lens

Control

Main Input

PIP/PBP Input

PIP / PBP ☐ ON ☒ OFF

Main Layout

Size

Swap

Test Pattern

Projector Info

Power OFF

Pic Mute ON

OSD OFF

IP Address 10.100.101.202

MAC Address 00:50:41:8a:14:65

Switch

Power

Pic Mute

OSD

Inspektor

Konsole

Debugger

Netzwerkanalyse

Stilbearbeitung

Laufzeitanalyse

Speicher

Web-Speicher

Barrierefreiheit

Anwendung

Adressen durchsuchen

Alles HTML CSS JS XHR Schriften Grafiken Medien WebSockets Sonstiges

Cache deaktivieren

Keine Drosselung

37 Anfragen

350,81 kB / 352,67 kB übertragen

Beendet: 16,47 s

DOMContentLoaded: 423 ms

load: 775 ms

Kopfzeilen

Cookies

Anfrage

Antwort

Zeit

Aufrufliste

In Warteschlange aufgenommen: 12,40 s | Gestartet: 12,40 s | Heruntergeladen: 13,16 s

Anfrage-Zeiten

Blockiert: 0 ms

DNS-Auflösung: 0 ms

Verbindungsaufbau: 1 ms

TLS-Konfiguration: 0 ms

Senden: 0 ms

Warten: 743 ms

Empfangen: 21 ms



Abhängigkeiten der Prozesse

(Ausgabe pstree)

```
systemd-+-ModemManager---2*[{ModemManager}]
|-applet.py
|-avahi-daemon---avahi-daemon
|-bluetoothd
|-cron---cron---sh---start-ola.sh---olad---7*[{olad}]
|-hciattach
|-lightdm-+-Xorg---{Xorg}
|   |-lightdm-+-lxsession-+-lxpanel---4*[{lxpanel}]
|   |   |   |   |-lxpolkit---2*[{lxpolkit}]
|   |   |   |   |-openbox
```

Auflistung der Prozesse

(Ausgabe top)

PID	USER	PR	NI	VIRT	RES	SHR	S	%CPU	%MEM	ZEIT+	BEFEHL
5460	fft	20	0	11464	3128	2552	R	22,2	0,2	0:00.07	top
1129	fft	20	0	10484	7732	7236	S	5,6	0,4	144:20.70	ola_trigger
463	fft	20	0	7768	1664	1544	S	0,0	0,1	0:00.00	start-ola_trigg
471	fft	20	0	7768	1600	1480	S	0,0	0,1	0:00.00	start-ola_trigg
474	fft	20	0	7768	1724	1604	S	0,0	0,1	0:00.00	start-ola_trigg
476	fft	20	0	1976	392	336	S	0,0	0,0	0:00.01	sh
483	fft	20	0	7768	1664	1544	S	0,0	0,1	0:00.00	start-ola_trigg
484	fft	20	0	7768	1600	1480	S	0,0	0,1	0:00.00	start-ola_trigg
491	fft	20	0	7768	2780	2628	S	0,0	0,1	0:00.01	start-ola.sh
494	fft	20	0	7768	1664	1544	S	0,0	0,1	0:00.00	start-ola_trigg
523	fft	20	0	7768	1640	1520	S	0,0	0,1	0:00.00	start-ola_trigg
651	fft	20	0	14548	7568	6568	S	0,0	0,4	0:00.28	systemd
652	fft	20	0	37208	2556	1496	S	0,0	0,1	0:00.00	(sd-pam)

Weiter Einsatzmöglichkeiten

- Alle Geräte mit integrierter Webschnittstelle lassen sich ansteuern
- mit Hilfe von liblo lassen sich OSC Befehle versenden
- schalten der GPIO Pins des RasPi (z.B. Relais)
- steuern von Mediaplayern
- ...der Phantasie sind da keine Grenzen gesetzt

Noch offen

- Stresstest

Alternativen

- Node-RED, von der IBM entwickelt für IOT Geräte, grafische Entwicklungsoberfläche



Stabilität

Ausgabe von uptime:

12:35:27 up 342 days, 18:03, 3 users, load average: 0,00, 0,00, 0,00

Fazit

- kostengünstige Lösung mit hoher Flexibilität und Stabilität
- Simple Einrichtung bei Produktionen (Ethernet)
- die verwendete Software ist gut dokumentiert und "abgehangen"
- ohne Vorkenntnisse ist eine Einarbeitung nötig
- Zeitaufwand für Implementierung und Test

Weblinks

https://www.johannes-felber.at/post/2019/05/beamer_control

<https://www.openlighting.org/ola/>

<https://www.raspberrypi.com/>

<https://github.com/frank-fft/Projektor-Shutter>

