Display Panel

Install

Wegen zu vielen Problemen mit Wayland wird hier **Trixie mit X11** genutzt.

Das Hauptproblem ist, dass das Display beim Stromsparen einen nervigen Bildschirm anzeigt … “No Signal” …

Einstellungen

- '' sudo raspi-config''
  - I2C Aktivieren
  - 1Wire aktivieren
  - umstellen von Wayland auf X11
  - Screen Blanking aus !

100 Mbit Fix

- Man muss dafür per SSH mit WLAN auf dem Pi sein ….
- `sudo nano /etc/systemd/system/eth0fix.service`
  - ```
    [Unit]
    Description=100MBit Fix for eth0
    After=network-online.target
    Requires=network-online.target

    [Service]
    ExecStart=/sbin/ethtool -s eth0 speed 100 duplex full autoneg on
    Type=oneshot

    [Install]
    WantedBy=multi-user.target
    ```
- `sudo systemctl enable eth0fix.service`
- `sudo systemctl restart eth0fix.service`

Updates

- Command Line
  `sudo apt update && sudo apt upgrade -y && sudo apt install -y git git-lfs silversearcher-ag wavemon hexedit sudoku tcpdump iptraf mc htop dcfldd nano usbutils ranger ncdu can-utils multitail fd-find lsof x11vnc`
    ```

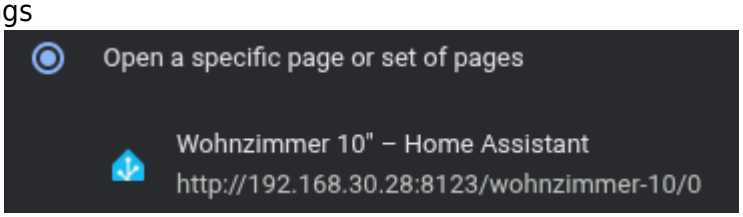
```
minicom joystick i2c-tools speedtest-cli iotop fio ir-keytable curl inxi
lshw nala fzf duf
```

- UI Tools
  ```
  sudo apt update && sudo apt upgrade -y && sudo apt install -y flameshot
  terminator cutecom jstest-gtk xclip
  ```
- Nutzloses Zeug
  ```
  sudo apt autoremove -y modem* cups* avahi* triggerhappy* rpi-connect
  openvpn xscreensaver
  ```

ser2net

- Der Zigbee Stick hängt am Display über USB. Damit der in HA genutzt werden kann nutzen wir ser2net.
- → https://www.drklipper.de/doku.php?id=haussteuerung:zigbee2mqtt

Browser

- Settings
  - 

    http://192.168.30.28:8123/wohnzimmer-10/0
    ⚠ Damit ist dann auch das “Restore Previous” weg beim Start!
  - Appearance → Mode → Dark

… steuern

falls benötigt …

```
# Chromium in Fullscreen
sleep 15
#WID=$(DISPLAY=:0 xdotool search --onlyvisible --class chromium|head -1)
#DISPLAY=:0 xdotool windowactivate ${WID}
#DISPLAY=:0 xdotool key F11

# An Display Position navigieren mit der Maus und Vollbild auslösen
DISPLAY=:0 /usr/bin/xdotool mousemove --sync 954 200 click 1
```

Autostart

- ⚠ Alles in die Autostart packen ist keine gute Option. xset wird nicht sauber gestartet.
- `nano /home/pi/start_kiosk.sh`

start_kiosk.sh

```
#!/bin/bash

# Warte kurz, bis der Desktop vollständig geladen ist (WICHTIG!)
sleep 5

# Energieeinstellungen deaktivieren
xset s off
xset s noblank
xset -dpms

# Chromium Crash-Flags bereinigen
# 1. Sperrdateien löschen (WICHTIG!)
# Verhindert die Meldung "Profil wird bereits verwendet"
rm -f /home/pi/.config/chromium/SingletonLock
rm -f /home/pi/.config/chromium/Default/SingletonLock

# 2. 'Local State' säubern (Globaler Status)
sed -i 's/"exited_cleanly":false/"exited_cleanly":true/'
/home/pi/.config/chromium/'Local State'

# 3. 'Preferences' säubern (Profil Status)
# Dieser eine Befehl erschlägt "Crashed", "SessionCrashed" und
andere Zustände auf einmal
sed -i
's/"exit_type":"[^"]\+"/"exit_type":"Normal"/;s/"exited_cleanly":f
alse/"exited_cleanly":true/'
/home/pi/.config/chromium/Default/Preferences

# Chromium starten (Kiosk Modus als Beispiel, ansonsten deine
Flags nutzen)
chromium --noerrdialogs --touch-events=enabled
```

- Unter Debian Trixie (X11) liegt die autostart dann hier :
`sudo nano /etc/xdg/lxsession/rpd-x/autostart`

autostart

```
@lxpanel-pi
@pcmanfm-pi
@bash /home/pi/start_kiosk.sh
```

Sonst liegt die Autostart hier `/home/pi/.config/lxsession/LXDE-pi/autostart`

- `chmod +x /home/pi/start_kiosk.sh`
- Nach einem Restart kann man `xset -q` prüfen ob Screenblanking und Screensaver Timeout deaktiviert sind:

```
pi@SHome-Display:~ $ xset -q
...
Screen Saver:
  prefer blanking:  no    allow exposures:  yes
  timeout:  0    cycle:  600
...
```

X11VNC

- `sudo apt install x11vnc`
- für einen ersten Test kann man das verwenden
  `x11vnc -usepw -forever -display :0`

- **Einrichtung als Dienst**
- `sudo x11vnc -storepasswd /etc/x11vnc.pass`
- `sudo nano /lib/systemd/system/x11vnc.service`

- ```
  [Unit]
  Description=Start X11VNC
  After=multi-user.target

  [Service]
  Type=simple
  ExecStart=/usr/bin/x11vnc -display :0 -auth guess -forever -loop -
  noxdamage -repeat -rfbauth /etc/x11vnc.pass -rfbport 5900 -shared

  [Install]
  WantedBy=multi-user.target
  ```

- `sudo systemctl enable x11vnc.service`

Fullscreen

- für Fullscreen Kontroller brauchen wir die HA Integration “Browser Mod” (Installieren über HACS)
  https://github.com/thomasloven/hass-browser_mod
- auf dem Endgerät dann die Seite öffnen
  http://192.168.30.28:8123/browser-mod
  - Register aktivieren
  - sinnvollen Namen in Browser ID eingeben `WZ_Display_10Zoll`
- Jetzt sollte der Browser als Device in HA auftauchen
- Wir brauchen BroiwserMod um JavaScript auszuführen um Fullscreen Toggel hinzubekommen
- Das Ausblenden von HA Elementen passiert mit `kiosk-mode`
  https://github.com/NemesisRE/kiosk-mode
- Konfig vom kiosk-mode passiert im YAML Modus des Dashboards!

```
kiosk_mode:
  user_settings:
```
```

```
    - users:
        - wohnzimmer
      hide_sidebar: true
      hide_header: false
      hide_assistant: true
```

Das muss am Anfang ergänzt werden. Damit wird für den User `wohnzimmer` die Sidebar und der Assistant abgeschaltet.

- Den Fullscreen Button kann man so erstellen:
  - Button mit Interaktion → Aktion ausführen → javascript
  - Target Browser auswählen → WZ Display (dafür braucht man browser_mod)
  - Code :

```
if (!document.fullscreenElement) {
  document.documentElement.requestFullscreen();
} else {
  document.exitFullscreen();
}
```

Display On / Off

Das Display kann über den GPIO Pin 36 vom Raspberry Pi an- und abgeschaltet werden. Das geht über ein einfaches toggeln. Am Ende macht das Python Script (gestartet als Service) aber noch einiges mehr...

```
import RPi.GPIO as GPIO
import time

# GPIO-Setup (BOARD-Modus für physische Pin-Nummern)
GPIO.setmode(GPIO.BOARD)
GPIO.setwarnings(False)

# Pin 36 (GPIO 16) für Display ON/OFF
GPIO.setup(36, GPIO.OUT)

def toggleOnOff():
    GPIO.output(36, GPIO.HIGH)
    time.sleep(0.4)
    GPIO.output(36, GPIO.LOW)
    print("Display toggled!")

# Toggle ausführen
toggleOnOff()
```

Funktionen

- Display an / aus
- Raspberry in HA mittels Autodiscover anmelden
- Schalten des Displays über HA
- Melden von Bewegung an HA (MQTT)
- einfaches Systemn Monitoring (vor allem Temperatur)

Venv einrichten

- `sudo apt -y update && sudo apt-get install -y build-essential cmake make gcc pkg-config python3-dev python3-virtualenv libsystemd-dev libopenblas-dev python3-setuptools`
- `cd ~ && mkdir -p DisplayHA && cd DisplayHA`
- `python -m venv . && source bin/activate`
- `pip3 install --upgrade pip && pip3 install RPi.GPIO pynput smbus paho-mqtt psutil setuptools`
- Compile VL53 Lib ...
  - `sudo apt-get install build-essential python-dev`
  - `cd ~ && git clone https://github.com/pimoroni/VL53L0X_rasp_python.git && cd VL53L0X_rasp_python`
  - `pip install .`
  - VL53L0X.py in den Scriptordner kopieren ...
    `cp ~/VL53L0X_rasp_python/python/VL53L0X.py ~/DisplayHA/`
- Tests starten ...

Motion Detect

- Sensor : VL53L0X

Service

- Aktivieren
  ```
  sudo systemctl enable display.service
  sudo systemctl daemon-reload
  sudo systemctl start display.service
  ```
- Log ansehen
  `journalctl -u display.service -f`
- `sudo nano /lib/systemd/system/display.service`

```
[Unit]
Description=Smart Display Controller (venv/User Mode)
# Wir warten, bis das Netzwerk und die Grafikoberfläche da sind
After=network-online.target graphical.target
Wants=network-online.target

[Service]
# --- User Context ---
```

```
User=pi
Group=pi

# --- Arbeitsverzeichnis ---
# Setzt das Verzeichnis, damit relative Pfade funktionieren
WorkingDirectory=/home/pi/DisplayHA

# --- Environment ---
# Diese Variablen sind wichtig für DDC/CI oder xrandr Interaktionen
Environment=DISPLAY=:0
Environment=XAUTHORITY=/home/pi/.Xauthority
# Erzwingt unbuffered Output, damit Logs sofort im Journal erscheinen
Environment=PYTHONUNBUFFERED=1

# --- ExecStart: Der Trick für das venv ---
# Statt 'source bin/activate' rufen wir direkt das Python im venv auf.
# PASSE DEN PFAD ZUM VENV AN, falls dein Ordner anders heißt (z.B.
.venv)!
ExecStart=/home/pi/DisplayHA/bin/python
/home/pi/DisplayHA/SmartDisplay.py

# --- Restart Verhalten ---
Restart=always
RestartSec=10s

[Install]
WantedBy=graphical.target
```

From:
https://drklipper.de/ - **Dr. Klipper Wiki**

Permanent link:
https://drklipper.de/doku.php?id=haussteuerung:display_panel

Last update: **2025/12/27 08:11**