

OnAirScreen

Bedienungsanleitung

Version 1.0.10

Sascha Ludwig · astrastudio.de

<https://www.astrastudio.de/onairscreen/>

Inhaltsverzeichnis

1. Überblick	3
2. Installation und Start	4
Windows	4
macOS	4
Linux (Debian / Ubuntu / Raspberry Pi OS)	4
Fedora	5
Raspberry-Pi-SD-Karten-Image	5
Erster Start	6
3. Hauptbildschirm	7
Bereiche im Detail	7
Vollbildmodus	8
4. Tastaturkürzel (Hotkeys)	9
Anwendung	9
Status-LEDs	9
AIR-Timer	9
OAS-USB-Tastatur (Sonderbelegung)	10
5. Einstellungsdialog	11
Schaltflächen (unten)	11
5.1 General	11
5.2 Updates	14
5.3 Network	15
5.4 Links	17
5.5 Advanced	19
5.6 Timers	20
5.7 Fonts	20
5.8 Time Source	21
5.9 Audio Meters	23
5.10 GPIO / MIDI	25
5.11 About	31
6. Funktionen im Detail	32
6.1 Status-LEDs	32
6.2 AIR-Timer	32
6.3 Textzeilen NOW, NEXT, WARN	33
6.4 IP-Adressen anzeigen	34
6.5 Uhr	34
6.6 Systembefehle	39
6.7 GPIO-Eingänge	40
7. Fernsteuerung und API	41
7.1 UDP (Port 3310)	41
7.2 HTTP (Port 8010)	41
7.3 Web-UI	41
7.4 REST-API	42
7.5 MQTT	43
7.6 Bitfocus Companion (empfohlen)	44
7.7 OSC (Port 8000)	45
7.8 SNMP (Port 1161)	46

7.9 Befehlsreferenz	47
8. Presets (Profile)	50
9. Kommandozeilenoptionen	51
10. Konfigurationsspeicherort	52
11. Fehlerbehebung	53
OnAirScreen startet nicht / Port belegt	53
Fernsteuerung funktioniert nicht	53
GPIO schaltet LEDs oder Timer nicht	53
Companion-Modul zeigt Disconnected	53
Silence Detection löst nicht aus	54
MQTT-Verbindung schlägt fehl	54
NTP-Warnung erscheint dauerhaft	54
Wetter-Widget zeigt nichts	54
Livewire-Meter zeigt keine Pegel	54
AES67-Meter: keine Streams oder keine Pegel	55
Einstellungen zurücksetzen	55
Logs an den Support schicken	55
Anhang: Ereignisprotokollierung	56

1. Überblick

OnAirScreen ist eine vielseitige **On-Air-Lampe** für professionelle Broadcast-Umgebungen. Die Anwendung kombiniert:

- **4 Status-LEDs** (ein-/ausschaltbar, blinkend, zeitgesteuert)
- **4 AIR-Timer** (Mikrofon, Telefon, Radio-Timer, Stream-Timer); AIR3 mit ▲/▼ und Top-of-Hour
- **Digitale oder analoge Uhr** mit optionalem Textuhr-Modus und Lock-LED (Local / NTP / PTP / LTC)
- **Stereo-Audio-Meter** (lokaler Eingang, Axia Livewire oder AES67 AoIP): L/R, Programme-LUFS oder beides, optional I+LRA
- **Silence Detection** mit optionalem WARN auf dem Bildschirm und Boolean in API/MQTT
- **Instanzname** (DNS-sicher, z. B. Studio-1) zur Unterscheidung mehrerer OnAirScreens
- **Textzeilen** NOW, NEXT und WARN (mit Prioritätssystem)
- **Wetter-Widget** (OpenWeatherMap)
- **Fernsteuerung** per Tastatur, Maus (Doppelklick/Rechtsklick/Lang-Klick), UDP, HTTP, Web-UI, MQTT, OSC, SNMP, REST-API und Bitfocus Companion
- **Home-Assistant-Integration** via MQTT Autodiscovery
- **GPIO-Eingänge** auf dem Raspberry Pi (Mixer-GPI über Optokoppler), **Axia-Livewire-GPIO** (Multicast oder LWRP TCP 93, z. B. Mic-ON-Lampe) und **MIDI** (Note/CC von einem Interface)
- **Instanz-Links**: 1→n-Replicas (voller LED/AIR/NOW/NEXT/WARN-Status) und Signal-Mappings (z. B. Flur-Tally von mehreren Studios)

Die Anwendung startet standardmäßig im **Vollbildmodus** mit ausgeblendetem Mauszeiger und eignet sich für dedizierte Studio-Monitore, Raspberry-Pi-Setups und Touch-freie Bedienung.

OnAirScreen passt sich automatisch an verschiedene Monitor-Seitenverhältnisse an und funktioniert sowohl auf **4:3**- als auch auf **16:9/16:10**-Displays.

2. Installation und Start

Fertige Builds für Windows, macOS, Linux und Raspberry Pi gibt es im [astrastudio-Shop](#). Eine Lizenz gilt für einen PC oder einen Raspberry Pi. Jeder weitere Rechner braucht eine eigene Lizenz. Dieselbe Lizenz kann auf einen Ersatzrechner übertragen werden, wenn der bisherige nicht mehr genutzt wird.

OnAirScreen startet im **Vollbild** mit ausgeblendetem Mauszeiger. Einstellungen: Ctrl+S (macOS: Cmd+S). Änderungen gelten erst nach **Apply**.

Windows

1. OnAirScreen_*_Win_x64.msi herunterladen (in OnAirScreen_Win_x64.zip).
2. Die MSI doppelklicken und dem Installer folgen. Die rechnerweite Installation braucht Administratorrechte.
3. OnAirScreen liegt unter C:\Program Files\OnAirScreen.
4. Start über das **Startmenü** oder das Desktop-Symbol.

Im Startmenü-Ordner liegen außerdem **OAS Send** (Befehlssender) und **OAS Send (no console)**. Einstellungen stehen in der Windows-Registry, Logs unter %LOCALAPPDATA%\astrastudio\OnAirScreen\logs\.

Eine ältere portable .exe wird vom MSI **nicht** entfernt. Die Datei nach der Installation selbst löschen.

Windows gleicht die Internetzeit standardmäßig nur etwa einmal pro Woche ab. Bei Zeitquelle **Local** kann die Uhr falsch gehen. Sync-Intervall im Betriebssystem verkürzen oder in OnAirScreen NTP / PTP / LTC nutzen (siehe 5.8 Time Source).

macOS

1. OnAirScreen_*_macOS.dmg herunterladen.
2. Das DMG öffnen und **OnAirScreen.app** nach **Programme** (Applications) ziehen. Nicht direkt aus dem DMG starten.
3. OnAirScreen aus Programme, Launchpad oder Spotlight starten.

Shop-Builds sind mit einer Apple Developer ID signiert und notariert. Nach dem Kopieren nach Programme startest du die App aus Programme, Launchpad oder Spotlight. Für lokale Audio-Meter die **Mikrofon**-Berechtigung erteilen.

Linux (Debian / Ubuntu / Raspberry Pi OS)

Offizielle Builds sind ein .deb. Nach der Installation liegt die App in /opt/onairscreen, startet als onairscreen und erscheint im Anwendungsmenü.

Shop-ZIP	Paket darin	Architektur	Für
OnAirScreen_Linux_x64.zip	OnAirScreen_*_Linux_x64.deb	amd64	Debian/Ubuntu-PCs
OnAirScreen_PiOS_x64.zip	OnAirScreen_*_RaspberryPiOSx64.deb	arm64	Raspberry Pi OS 64-bit

Prüfen mit `dpkg --print-architecture`. Die beiden Pakete sind nicht austauschbar.

Raspberry Pi: 64-bit-OS auf Zero 2 W, Pi 3 / 4 / 400, CM4, Pi 5 / 500, CM5. Nicht unterstützt: Pi 1 und klassisches Zero. Es braucht eine grafische Desktop-Sitzung, kein Lite ohne GUI.

ZIP entpacken, danach mit **apt** installieren (nicht allein `dpkg -i`), damit PortAudio, OpenGL und Xcb nachgezogen werden. Das `./` vor der Datei ist Pflicht:

```
unzip OnAirScreen_Linux_x64.zip
cd ~/Downloads
sudo apt update
sudo apt install ./OnAirScreen_*.deb
```

Start über das Anwendungsmenü oder `onairscreen`. Update: neues `.deb` ebenso installieren; Einstellungen bleiben. Deinstallieren: `sudo apt remove onairscreen` (Config unter `~/.` bleibt).

Falls `dpkg -i` Abhängigkeiten vermisst hat: `sudo apt -f install`.

Pakete sind OpenPGP-signiert. Optionale Prüfung: [Package signing \(GPG\)](#).

Fedora

Das Fedora-RPM wird auf Fedora 44 gebaut (Fedora 44+). Ältere RHEL-, Rocky- oder Alma-Versionen können wegen `glibc` scheitern.

```
sudo rpm --import https://www.astrastudio.de/GPG-KEY-astrastudio
sudo dnf install ./OnAirScreen_*_Fedora_x64.rpm
```

`dnf` zieht PortAudio und weitere Systembibliotheken nach. Signatur prüfen: `rpm -K OnAirScreen_*_Fedora_x64.rpm` (erwartet: `digests signatures OK`).

Raspberry-Pi-SD-Karten-Image

Das [SD-Karten-Image](#) ist ein startfertiges 64-bit-Raspberry-Pi-OS mit installiertem OnAirScreen (Desktop-Autologin, GPIO inklusive). ZIP entpacken; darin liegt `OnAirScreen_*_RaspberryPiOS_Image.img`.

[Raspberry Pi Imager](#) verwenden:

- [macOS](#)
- [Windows](#)
- [Ubuntu x86](#)

1. microSD-Karte einlegen.
2. Raspberry Pi Imager starten.
3. **OS wählen** → **Eigenes Image verwenden** (Use custom) → die OnAirScreen-.img wählen (nicht die .zip).
4. **Speicher wählen** (bzw. **SD-Karte wählen**) und die Karte auswählen.
5. **Schreiben**.

Standardbenutzer: pi / Passwort OnAirScreen1!. Hostname: onairscreen. Passwort nach dem ersten Start ändern. SSH ist eingeschaltet. Host-Reboot und Shutdown aus OnAirScreen (Einstellungen, Web-UI oder CMD:REBOOT / CMD:SHUTDOWN) funktionieren auf diesem Image ohne Passwort.

Auf dem Desktop: **OnAirScreen, OAS Pi Settings** (optionaler Autostart), Bildschirmtastatur, Netzwerk-Symbol. Beim ersten Boot wächst das Root-Dateisystem auf die Kartengröße.

Wer bereits Raspberry Pi OS 64-bit mit Desktop nutzt, installiert stattdessen das Pi-.deb (siehe Linux oben).

Erster Start

Beim ersten Start werden Standardeinstellungen geladen. Der Einstellungsdialog öffnet sich mit Ctrl+S (macOS: Cmd+S). Änderungen werden erst nach Klick auf **Apply** übernommen und gespeichert.

3. Hauptbildschirm

Der Hauptbildschirm ist in folgende Bereiche gegliedert:

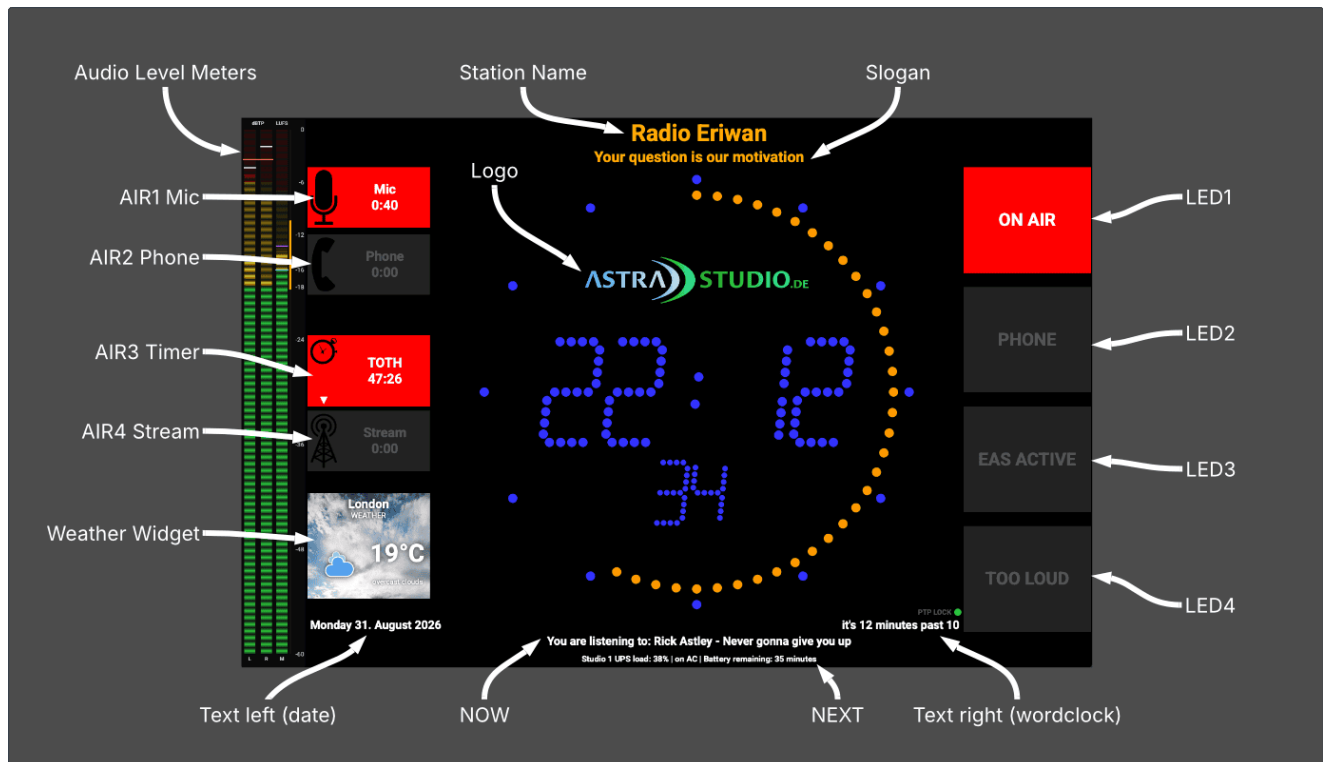


Abbildung 1: OnAirScreen Hauptbildschirm, beschriftet

Bereiche im Detail

Bereich	Widget	Funktion
Stationsname	labelStation	Name des Senders, Farbe konfigurierbar
Slogan	labelSlogan	Untertitel / Claim des Senders
Status-LEDs 1-4	buttonLED1-buttonLED4	Große farbige Statusanzeigen (ON AIR, PHONE, ...)
Audio-Meter	audioMeterWidget	L/R und/oder Programme-LUFS am linken Rand (optional)
Uhr	clockWidget	Digital oder analog, mit Logo und optionalem Wetter-Widget
Lock-LED	Clock lock	Unten rechts: PTP/NTP/LTC LOCK oder LOCAL
AIR-Timer 1-4	AirLED_1-AirLED_4	Stoppuhr-Timer mit Icon, Label, MM:SS; AIR3 mit ▲/▼
Datum	labelTextLeft	Unten links: Wochentag und Datum
NOW	labelCurrentSong	Erste Fußzeile (z. B. aktueller Songtitel)
NEXT	labelNews	Zweite Fußzeile (z. B. nächster Titel)
WARN	labelWarning	Warnmeldung; blendet NOW/NEXT aus, wenn aktiv
Textuhr	labelTextRight	Unten rechts: Uhrzeit in Worten (Wordclock)

Hinweis: Status-LEDs und AIR-Timer lassen sich per **Linksklick**, **Tastatur** oder **Fernsteuerung** umschalten. Doppelklick auf LED/Timer ändert den Vollbildmodus nicht.

Vollbildmodus

- Standard: Vollbild mit verstecktem Mauszeiger
 - Umschalten: F oder Ctrl+F (macOS: Cmd+F), **Doppelklick** auf eine leere Fläche des Hauptbildschirms, oder Rechtsklick / Lang-Klick → **Toggle Fullscreen**
 - **Rechtsklick-/Lang-Klick-Menü:** Toggle Fullscreen, Settings, Hotkeys, Quit OnAirScreen; bei aktiven Metern zusätzlich **Start I+LRA**, **Stop I+LRA** und **Reset I+LRA**
 - Der Vollbild-Zustand wird in den Einstellungen (General/fullscreen) gespeichert
 - Einstellungen → General → **Always start in fullscreen mode** (General/always_start_fullscreen, Standard aus) startet immer im Vollbild, auch wenn die letzte Session im Fenster war. Apply gilt erst beim nächsten Start.
 - Im Fenstermodus werden Position und Größe gespeichert (Window/geometry) und beim nächsten Start wiederhergestellt
-

4. Tastaturkürzel (Hotkeys)

Auf **macOS** wird Ctrl durch Cmd (⌘) ersetzt.

Anwendung

Taste(n)	Funktion
H	Hotkey-Overlay ein/aus
F / Ctrl+F	Vollbild ein/aus
Ctrl+S / Ctrl+,	Einstellungsdialog öffnen
Q / Ctrl+Q / Ctrl+C / ESC	OnAirScreen beenden
I	IP-Adressen für 10 Sekunden in NOW/NEXT anzeigen

Das Overlay listet diese Tastenkürzel auf dem Hauptbildschirm. Schließen mit H, Esc oder einem Klick auf den abgedunkelten Hintergrund. Andere Main-Window-Hotkeys sind gesperrt, solange es offen ist; Esc beendet OnAirScreen erst, wenn das Overlay geschlossen ist. Rechtsklick / Lang-Klick → **Hotkeys** öffnet dasselbe Overlay.

Beim Beenden erscheint die WARN-Meldung QUITTING ONAIRSCREEN auf dem Hauptbildschirm, bis das Fenster schließt.

Status-LEDs

Taste	Funktion
1	LED 1 ein/aus
2	LED 2 ein/aus
3	LED 3 ein/aus
4	LED 4 ein/aus

Linksklick auf eine Status-LED schaltet sie ebenso um.

AIR-Timer

Taste(n)	Funktion	Timer
M / /	Start/Stopp	AIR1 (Mikrofon)
P / *	Start/Stopp	AIR2 (Telefon)
Leertaste / , / .	Start/Stopp	AIR3 (Radio-Timer)
S	Start/Stopp	AIR4 (Stream-Timer)
0 / R	Zurücksetzen auf 0:00	AIR3 (Radio-Timer)
Alt+S	Zurücksetzen auf 0:00	AIR4 (Stream-Timer)
T	Top-of-Hour-Countdown ein/aus	AIR3
Enter / Return	Timer-Eingabedialog öffnen	AIR3

Linksklick auf einen AIR-Timer startet oder stoppt ihn (wie M / P / Leertaste / S). Reset, TOTH und Timer-Dialog bleiben Tastatur bzw. Fernsteuerung.

OAS-USB-Tastatur (Sonderbelegung)

Taste	Funktion
Display-Taste	Vollbild ein/aus
Calculator-Taste	Host herunterfahren (shutdown_host)

5. Einstellungsdialog

Der Einstellungsdialog öffnet sich mit **Ctrl+S** oder **Ctrl+,** (oder Rechtsklick / Lang-Klick → **Settings**). Ist er bereits offen, kommt das Fenster nach vorne, ohne die Werte neu zu laden. Laufende AIR-Timer (inkl. TOTH) bleiben dabei unverändert. Der Dialog enthält **mehrere Registerkarten** (vertikal links angeordnet):

Register	Inhalt
General	Instanzname, Station, LEDs, Uhr, Logo, Vollbild-Start
Updates	Update-Prüfung, Update Key, Hardware-Aktivierung
Network	UDP, HTTP, Multicast, MQTT, OSC, SNMP
Links	Andere OnAirScreens finden; Replicas (1→n voller Status) und Source-Mappings
Advanced	Formatierung, Wetter
Timers	AIR-Timer 1–4
Fonts	Schriftarten für alle Elemente
Time Source	Local / NTP / PTPv2 / LTC, NTP-Prüfung
Audio Meters	Pegelanzeige, Quelle, TooLoud, Silence Detection
GPIO	Raspberry Pi, Axia Livewire und MIDI (parallel)
About	Version, Lizenzinfo, Log-Level, Log-Ordner, Reset, Host-Reboot/Shutdown
License	OASL 1.0 und Third-Party-Hinweise (PySide6/Qt, Fonts, Beispiele)

Schaltflächen (unten)

Schaltfläche	Funktion
Quit	OnAirScreen beenden
Delete Preset...	Gespeichertes Preset löschen
Load Preset...	Preset laden
Save Preset...	Aktuelle Konfiguration als Preset speichern
Close	Dialog schließen ohne zu speichern (Änderungen verwerfen)
Apply	Alle Einstellungen übernehmen, speichern und Dialog schließen

5.1 General

Instanzname

Einstellung	Schlüssel	Standard	Beschreibung
Instance Name	General/ instancename	Studio-1	DNS-sicheres Label (1–32 Zeichen, Buchstaben/Ziffern/Bindestrich, kein führendes oder nachgestelltes –) zur Identifikation dieses OnAirScreens

Der Instanzname erscheint in der Web-UI (Titel und Status), in `/api/status (instance)`, auf MQTT `{base}/instance/state` und als Home-Assistant-Sensor. Der HA-Gerätename wird zu `OnAirScreen (Studio-1)`, sofern der Instanzname nicht schon im Device Name steht. Remote: `CONF:General:instancename=TEXT`.

Der Name muss im LAN eindeutig sein. Kündigt ein anderer OnAirScreen denselben Namen an, zeigen Settings → General, Settings → Links und die Web-UI eine Warnung mit den anderen Hosts. Nearby listet jeden kollidierenden Host extra (`name collision`). Linking rät die IP nicht – Host eintragen oder eine Instanz umbenennen. Apply wird abgelehnt, solange der Name kollidiert.

Stationsname und Slogan

Einstellung	Schlüssel	Standard	Beschreibung
Station Name	General/ stationname	Radio Eriwan	Name des Senders
Station Color	General/ stationcolor	#FFAA00	Textfarbe Stationsname
Slogan	General/slogan	Your question is our motivation	Slogan / Claim
Slogan Color	General/ slogancolor	#FFAA00	Textfarbe Slogan

Eine Live-Vorschau (`StationNameDemo`, `SloganDemo`) zeigt die Eingaben sofort an.

Status-LEDs 1–4

Für jede LED (Gruppen LED1–LED4):

Einstellung	Schlüssel	Standard	Beschreibung
Aktiviert	used	true	LED auf dem Hauptbildschirm anzeigen
Text	text	siehe unten	Beschriftung der LED

Einstellung	Schlüssel	Standard	Beschreibung
Active BG Color	activebgcolor	#FF0000	Hintergrundfarbe (aktiv)
Active Text Color	activetextcolor	#FFFFFF	Textfarbe (aktiv)
Autoflash	autoflash	false	Dauerblinker alle 500 ms
20sec flash	timedflash	false	20 Sekunden blinken, dann automatisch aus

Standard-LED-Texte:

LED	Text
LED1	ON AIR
LED2	PHONE
LED3	DOORBELL
LED4	EAS ACTIVE

Gemeinsame inaktive Farben (Gruppe LEDS):

Einstellung	Schlüssel	Standard
Inactive BG Color	inactivebgcolor	#222222
Inactive Text Color	inactivetextcolor	#555555

Logo

Einstellung	Schlüssel	Standard	Beschreibung
Logo Path	Clock/logopath	:/astrastudio_ logo/...	Pfad zum Logo-Bild
Logo Position	Clock/logoUpper	false (unten)	Logo oben oder unten in der Uhr

Schaltflächen: ... (Dateiauswahl), reset (Standard-Logo wiederherstellen).

OnAir Clock Mode / Colors

Einstellung	Schlüssel	Standard	Beschreibung
Clock Face	Clock/face	digital	digital, analog, analog_numbers, analog_studio, analog_railway, analog_24h_smooth, analog_24h_ticking (analog_24h ist Alias für Smooth)

Einstellung	Schlüssel	Standard	Beschreibung
Digital (Legacy)	Clock/digital	true	Wird mit Clock/face mitgeschrieben (true = Digital)
Hours LEDs	Clock/digitalhourcolor	#3232FF	Farbe Stunden-LEDs
Seconds LEDs	Clock/digitalsecondcolor	#FF9900	Farbe Sekunden-LEDs
Digits LEDs	Clock/digitaldigitcolor	#3232FF	Farbe aller Ziffern
Show seconds	Clock/showSeconds	false	Sekunden anzeigen
Seconds Layout	Clock/showSecondsInOneLine	false	separate oder in one line
Static colon	Clock/staticColon	false	Doppelpunkt statisch (nicht blinkend)
Use textclock	Clock/useTextClock	true	Textuhr (z. B. „it's 3 o'clock")
Replace IPs after 10s	General/replacenow	false	Nach IP-Anzeige Text ersetzen
Replace with text	General/replacenowtext	(leer)	Ersatztext für NOW-Zeile

Anzeige

Einstellung	Schlüssel	Standard	Beschreibung
Always start in fullscreen mode	General/always_start_fullscreen	false	Immer im Vollbild starten, auch wenn die letzte Session im Fenster war

Apply wechselt das aktuelle Fenster nicht; die Option gilt beim nächsten Start. Ohne sie stellt OnAirScreen weiterhin den letzten Vollbild-Zustand wieder her (General/fullscreen, Standard an).

5.2 Updates

Update-Prüfung (kompilierte Versionen)

Einstellung	Schlüssel	Standard	Beschreibung
Check for updates	General/updatecheck	false	Automatische Update-Prüfung beim Start
Update Key	General/updatekey	(leer)	Update-Schlüssel für kostenpflichtige Versionen (siehe Hinweis unten)
Include Beta Versions	General/updateincludebeta	false	Beta-Versionen einschließen

Die Update-Funktion ist für **vorkompilierte (kostenpflichtige) Versionen** gedacht.

Für die Update-Prüfung in der kostenpflichtigen Version muss ein **Update Key** eingegeben werden. Diesen erhältst du nach der Bestellung im Kundenportal unter customer.astrastudio.de. Eine Lizenz gilt für einen PC oder einen Raspberry Pi. Das Feld ist maskiert; mit dem durchgestrichenen Auge wird es sichtbar.

Aktivierung (kostenpflichtige Builds)

Shop-Builds müssen aktiviert werden. Selbst gebaute OpenSource-Builds nicht.

1. Update Key aus dem Kundenportal ins Feld **Update Key** kopieren.
2. **Activate** (mit Internet) oder die **Hardware ID** kopieren, im Portal eine Lizenzdatei laden und **Import license file**.
3. Die Aktivierung gilt ein Jahr, sofern die Bestellung nichts anderes sagt. Der Tab Updates zeigt, ob dieser Rechner jetzt online verlängern kann. Wenn der Lizenzserver nicht erreichbar ist, vor Ablauf im Portal eine neue Lizenzdatei laden und importieren. Der Update Key bleibt derselbe. Es ist kein neuer Kauf.
4. Zum Umziehen **Return license**. Den Return-Code im Portal einlösen (online erledigt das die App). Den neuen Rechner aktivieren. Ein defekter Rechner kann nur vom Support freigegeben werden.
5. Ohne Lizenz läuft OnAirScreen 14 Tage voll. Danach kein Vollbild, kein GPIO, keine Audio-Meter, keine gespeicherten Einstellungen, Fenster höchstens 1280×800, Ende nach 15 Minuten. Dasselbe gilt, wenn eine Lizenz abgelaufen ist.
6. Beim Start wird der Update Key beim Lizenzserver geprüft, auch wenn **Check for Updates** aus ist. Lehnt der Server den Key ab, wird die Lizenz auf diesem Rechner sofort entfernt und es gelten dieselben Grenzen. Weitere Starts wiederholen diese automatische Prüfung und die Meldung nicht. **Check now** prüft weiter. Eine fehlende Netzwerkverbindung entfernt die Lizenz nicht. Eine neue Aktivierung prüft den Key wieder.

Passt diese Aktivierung für deinen Einsatz nicht, schreib uns über das [Kontaktformular](#). Wir erarbeiten dann eine maßgeschneiderte Lösung.

Dieselben Aktionen gibt es in den Web-UI-Einstellungen.

5.3 Network

UDP / HTTP / Multicast

Einstellung	Schlüssel	Standard	Beschreibung
UDP Port	Network/udpport	3310	Port für UDP-Befehle
HTTP Port	Network/httpport	8010	Port für HTTP/Web-UI

Einstellung	Schlüssel	Standard	Beschreibung
Multicast Address	Network/multicast_address	239.194.0.1	Multicast-Adresse für UDP
Web Settings PIN	Network/websettingspin	(leer)	Optionale PIN für das Settings-Overlay der Web-UI und für HTTP CMD:REBOOT / CMD:SHUTDOWN (als Hash gespeichert). Leer = keine PIN. Feld leer lassen, um die aktuelle PIN zu behalten; – entfernt sie. Andere Remote-Control-Befehle bleiben ungeschützt.

MQTT

Einstellung	Schlüssel	Standard	Beschreibung
enable MQTT support	MQTT/enablemqtt	false	MQTT-Integration einschalten
MQTT Server	MQTT/mqttserver	localhost	Broker-Hostname/IP
MQTT Server Port	MQTT/mqttport	1883	Broker-Port
MQTT User	MQTT/mqttuser	(leer)	Benutzername (optional)
MQTT Password	MQTT/mqttpassword	(leer)	Passwort (optional)
MQTT Device Name	MQTT/mqttdevicename	OnAirScreen	Gerätename in Home Assistant

MQTT-Passwort und API-Keys sind maskiert; mit dem durchgestrichenen Auge werden sie sichtbar.

MQTT Base Topic: onairscreen + letzte 6 Hex-Zeichen der MAC-Adresse, z. B. onairscreen_a1b2c3.

OSC

Einstellung	Schlüssel	Standard	Beschreibung
enable OSC support	OSC/enableosc	false	OSC-Fernsteuerung einschalten
OSC Listen Port	OSC/oscport	8000	UDP-Port für eingehende OSC
OSC Send Host	OSC/oscsendhost	(leer)	Ziel für Status-Push (Companion-IP); leer = kein Push
OSC Send Port	OSC/oscsendport	9000	Ziel-UDP-Port (Companion-Feedback-Port)

Send Host wird nur für unaufgeforderten Status-Push gebraucht (Companion-Button-Feedback). Abfragen antworten immer an den UDP-Absender.

SNMP

Einstellung	Schlüssel	Standard	Beschreibung
enable SNMP support	SNMP/enablesnmp	false	SNMPv2c/v3-Agent einschalten
SNMP Listen Port	SNMP/snmpport	1161	UDP-Port. 161 braucht OS-Rechte
sysContact	SNMP/syscontact	(leer)	MIB-2 sysContact
sysLocation	SNMP/syslocation	(leer)	MIB-2 sysLocation
v2c Read Community	SNMP/snmpcommunity	public	Leer = v2c aus. Maskiert
v2c Write Community	SNMP/snmpwritecommunity	(leer)	Leer = v2c nur lesen. Maskiert
v3 User	SNMP/snmpv3user	(leer)	Leer = v3 aus
v3 Auth Password	SNMP/snmpv3authpass	(leer)	SHA-256. Leer = noAuthNoPriv (nur Test). Maskiert
v3 Priv Password	SNMP/snmpv3privpass	(leer)	AES-128. Leer = authNoPriv, wenn Auth gesetzt. Maskiert
enable SNMP traps	SNMP/enabletraps	false	Traps an das NMS senden
Trap Host	SNMP/snmptraphost	(leer)	NMS-Hostname/IP
Trap Port	SNMP/snmptrapport	162	NMS-Trap-Port
Trap Version	SNMP/snmptrapversion	v2c	v2c oder v3
Trap Community	SNMP/snmptrapcommunity	public	Community für v2c-Traps. Maskiert
trap on silence	SNMP/trap_silence	true	Trap bei Silence an/aus
trap on warning	SNMP/trap_warning	true	Trap bei WARN-Textwechsel
trap on LED change	SNMP/trap_leds	false	Trap bei LED an/aus (kann laut werden)

Mindestens eine v2c-Community oder ein v3-User muss gesetzt sein, sonst startet der Agent nicht. Community- und v3-Passwörter sind maskiert.

5.4 Links

Mehrere OnAirScreens im LAN (Multicast-Discovery über den **Instanznamen**) oder per Host/HTTP-Port verbinden. Instanznamen müssen eindeutig sein; eine Kollision wird in Settings und der Web-UI angezeigt, Apply wird abgelehnt. Es gibt keine festen Master/Slave-Rollen: jede Instanz kann Replicas pushen **und** Sources abonnieren. Runtime-Status wird bei Änderung gepusht, nicht per Dauer-HTTP-Poll an alle Peers.

Host-lokale WARN-Texte (QUITTING ONAIRSCREEN, Reboot/Shutdown, NTP) bleiben auf diesem Screen.

Replicas (dieser Screen ist die Quelle): Instanzen aus der Nearby-Liste hinzufügen. Apply schreibt Follow auf dem Ziel und schickt den kompletten **Runtime**-Status: LED1–4 (Zustand, Beschriftung, Autoflash/Timedflash), AIR1–4 (Zustand, Sekunden, ToH/Countdown), NOW / NEXT / WARN (inkl. Warn-Stack). Uhr, Meter, GPIO-Verdrahtung, Netzwerkports und der Instanzname bleiben lokal. 1→n wird einmal an der Quelle konfiguriert.

Sources (dieser Screen ist das Ziel): z. B. Studio-1 / LED1 → LED1 und Studio-2 / LED1 → LED2 für einen Flur-Tally. Signale: LED1–4, AIR1–4, NOW, NEXT, WARN, Silence oder **Follow all**. Follow all kopiert den kompletten Runtime-Status, Local target und Invert sind dann ohne Wirkung. Invert gilt für boolesche Signale. **Custom** sind Extra-API-Befehle (Semikolon), die feuern wenn die Quelle aktiv ist — zusätzlich zum Local target. Local target **Custom only** nutzt nur diese Befehle, ohne lokale LED/AIR/Text.

Nur **ein** Full-Follow pro Screen. Ist er schon Replica von Studio-1, kann Studio-2 ihn nicht zusätzlich als Replica eintragen (HTTP 409). Mappings anderer Instanzen bleiben für unbelegte Felder erlaubt. Offline-Quellen halten den letzten Zustand (kein automatisches Alles-AUS). Replica entfernen löscht Follow auf dem Ziel und lässt die Lampen stehen.

Discovery: Multicast 239.194.0.2 UDP **3311** (nicht die Befehlsgruppe 239.194.0.1:3310). HTTP /api/link/follow, /unfollow, /snapshot ohne Web-Settings-PIN (wie UDP/HTTP-Commands).

Einstellung	Schlüssel	Standard	Beschreibung
Enable instance linking	Links/enabled	false	Replicas pushen und Mappings anwenden. Namens-Discovery für Eindeutigkeit läuft immer.
Replica <i>n</i>	Links/replicaN_*	aus	Ziel-Instanz (Dropdown der Nearby-Screens, oder Name tippen). Host leer lassen, dann kommt die aktuelle IP vom Namen (DHCP-fest). Host nur als Fallback, wenn Multicast sie nicht erreicht, oder um bei Namenskollision eine Maschine zu wählen.
Map <i>n</i>	Links/mapN_*	aus	Source-Instanz (Nearby-Dropdown pro Zeile), Signal, lokales Ziel, Invert, Custom-Command. Host leer = per Name finden.
Replica of	Links/follow_instance	(leer)	Wird gesetzt, wenn ein anderer Screen diese Instanz als Replica einträgt

5.5 Advanced

Einstellung	Schlüssel	Standard	Beschreibung
Date format	Formatting/ dateFormat	dddd, dd. MMM yyyy	Qt-Datumsformat
Time format	Formatting/isAmPm	false (24h)	24-Stunden oder AM/PM
Textclock Language	Formatting/ textClockLanguage	English	Sprache der Textuhr

Verfügbare Textuhr-Sprachen: English, German, Dutch, French

Datumsformat-Platzhalter (Qt-Notation, Auszug):

Platzhalter	Bedeutung
d / dd	Tag (1–31 / 01–31)
ddd / dddd	Wochentag (kurz / lang)
M / MM	Monat (1–12 / 01–12)
MMM / MMMM	Monatsname (kurz / lang)
yy / yyyy	Jahr (2 / 4 Stellen)

Weather Widget (OpenWeatherMap)

Einstellung	Schlüssel	Standard	Beschreibung
show Weather Widget	WeatherWidget/ owmWidgetEnabled	false	Wetter-Widget aktivieren
API Key	WeatherWidget/ owmAPIKey	(leer)	OpenWeatherMap API-Schlüssel
City ID	WeatherWidget/ owmCityID	2643743 (London)	OpenWeatherMap City-ID
Language	WeatherWidget/ owmLanguage	English	Sprache der Wetterbeschreibung
Unit	WeatherWidget/ owmUnit	Celsius	Celsius, Fahrenheit oder Kelvin

Städtenamen neben **City ID** eingeben, **Find** (oder Return) drücken und einen Treffer in der Liste wählen. Die City-ID wird automatisch eingetragen; die ID kann weiterhin manuell gesetzt werden. Das Widget zeigt den gesuchten Stadtnamen, nicht den OpenWeatherMap-Stations- oder Stadtteilnamen. Der API-Key ist maskiert; mit dem durchgestrichenen Auge wird er sichtbar.

Test API: Schaltfläche zum Testen der API-Verbindung mit aktuellen Einstellungen.

Verfügbare Wetter-Sprachen: Arabic, Bulgarian, Catalan, Czech, German, Greek, English, Persian (Farsi), Finnish, French, Galician, Croatian, Hungarian, Italian, Japanese, Korean, Latvian, Lithuanian, Macedonian, Dutch, Polish, Portuguese, Romanian, Russian, Swedish, Slovak, Slovenian, Spanish, Turkish, Ukrainian, Vietnamese, Chinese Simplified, Chinese Traditional

Weitere Informationen: [WeatherWidget Guide](#)

5.6 Timers

Für jeden AIR-Timer (Gruppe Timers):

Einstellung	Schlüssel	Standard	Beschreibung
Aktiviert	TimerAIR{n}Enabled	true	Timer auf Hauptbildschirm anzeigen
Text	TimerAIR{n}Text	siehe unten	Label des Timers
Active BG Color	AIR{n}activebgcolor	#FF0000	Hintergrundfarbe (aktiv)
Active Text Color	AIR{n}activetextcolor	#FFFFFF	Textfarbe (aktiv)
Icon Path	air{n}iconpath	Standard-Icon	Pfad zum Timer-Icon

Standard-AIR-Bezeichnungen und Icons:

Timer	Standard-Text	Standard-Icon	Funktion
AIR1	Mic	Mikrofon-Icon	Mikrofon-Stoppuhr
AIR2	Phone	Telefon-Icon	Telefon-Stoppuhr
AIR3	Timer	Timer-Icon	Radio-Timer (Hoch-/Runterzählen, Top-of-Hour)
AIR4	Stream	Antennen-Icon	Stream-Timer

Einstellung	Schlüssel	Standard	Beschreibung
TOTH Timer Text	TimerTOTHText	TOTH Timer	AIR3-Beschriftung, solange der Top-of-Hour-Countdown aktiv ist
AIR Min Width	TimerAIRMinWidth	200	Mindestbreite der AIR-Anzeigen (Pixel)

5.7 Fonts

Für jedes UI-Element kann Schriftart, -größe und -stärke individuell gesetzt werden:

Element	Gruppe Fonts	Standard
LED1-4	LED{n}FontName/Size/Weight	Roboto, 32pt, Bold
AIR1-4	AIR{n}FontName/Size/Weight	Roboto, 24pt, Bold
Station Name	StationNameFontName/Size/Weight	Roboto, 24pt, Bold
Slogan	SloganFontName/Size/Weight	Roboto, 18pt, Bold

Die Familien-Combo listet Application Fonts (inkl. mitgeliefertem Roboto und Noto Sans). Die Größe ist ein Punkt-SpinBox (8–96 pt), **Bold** schaltet die Stärke, **Reset** stellt Roboto mit Default-Größe und Bold für die Zeile wieder her. Die Vorschau zeigt Beispieltext in der gewählten Schrift.

Zusätzlich werden Schriftarten aus dem fonts/-Verzeichnis beim Start geladen.

5.8 Time Source

Die Anzeigeuhr kann der lokalen Systemuhr, einem NTP-Server, einem PTPv2-Master (IEEE 1588-2008) oder SMPTE-LTC folgen (Leo-Bodnar-LBE-1110-USB-Serial **oder** Dekodierung von einem lokalen Audioeingang). **OnAirScreen ändert die Betriebssystem-Uhr nie.** NTP und PTP steuern eine eigene Zeitbasis (`time.monotonic()`); ein Sprung der Systemuhr bewegt die Studio-Uhr nicht. LTC darf springen, stehenbleiben und enthält Frames (HH:MM:SS:FF).

Datum, Textuhr und AIR3-Top-of-Hour folgen derselben Wandzeit wie die große Uhr, wenn die Quelle Local, NTP oder PTP ist. Bei LTC bleiben sie auf der Systemzeit, weil LTC kein Kalenderdatum hat.

Einstellung	Schlüssel	Standard	Beschreibung
Time Source	TimeSource/source	local	local, ntp, ptp oder ltc
Enable NTP-Check	NTP/ntpcheck	true	Warnung bei Abweichung vom NTP-Server oder wenn der Server nicht erreichbar ist
NTP Server	NTP/ntpcheckserver	pool.ntp.org	NTP-Server für NTP als Zeitquelle und für die optionale NTP-Prüfung
PTP Interface	TimeSource/ptp_iface	(leer)	IPv4-Adresse des PTP-Netzwerkinterfaces (unabhängig von AoIP)
PTP Domain	TimeSource/ptp_domain	0	IEEE-1588-Domain (0–255)
LTC Input	TimeSource/ltc_input	serial	serial (LBE-1110) oder audio (lokale PortAudio-Dekodierung)
LTC Serial Port	TimeSource/ltc_port	(leer)	USB-Serial-Gerät des LBE-1110; leer = Auto

Einstellung	Schlüssel	Standard	Beschreibung
LTC Audio Device	TimeSource/ltc_audio_device	(<i>leer</i>)	Lokales Capture-Gerät; leer = System-Default
LTC Channel	TimeSource/ltc_audio_channel	0	0 = Links, 1 = Rechts
LTC-Unlock-Warnung	TimeSource/ltc_warn	false	WARN-Text bei LTC-Verlust; die LED LTC NOT LOCKED bleibt immer aktiv

Local System Clock: Anzeige = Systemuhr. Mit NTP-Check wird die Systemuhr mit dem NTP-Server verglichen (Abweichung > 0,3 s oder Fehler → Warnung, Priorität -1).

NTP Server: OnAirScreen fragt das Feld NTP Server ab und steuert eine eigene Uhr. Nach dem ersten gültigen Sample wird die OS-Zeit nicht mehr verwendet. Das Feld bleibt editierbar, auch wenn NTP-Check aus ist. Bei NTP-Verlust läuft die letzte Zeit weiter, plus Warnung.

PTPv2 IEEE 1588–2008: Software-Slave auf Multicast 224.0.0.1:129 UDP 319/320, Delay-Mechanismus E2E. Typische Genauigkeit im Millisekundenbereich (kein Hardware-Timestamping). Das Interface unabhängig vom AoIP-Interface der Audio Meters wählen. Bei Sync-Verlust: letzte Zeit läuft weiter, plus Warnung. Wenn UDP 319/320 nicht gebunden werden kann (Ports schon belegt), versucht der Slave alle paar Sekunden erneut; Lock-LED und WARN zeigen PTP ERROR, bis die Ports frei sind. Details stehen im Log. Unter Linux und Raspberry Pi OS installieren Shop- .deb/ .rpm/SD-Images den Helper /usr/lib/onairscreen/oas-ptp-bind für UDP 319/320.

LTC: Eine Zeitquelle mit zwei Eingängen. Die Framerate wird aus den eingehenden Frames abgeleitet (24 / 25 / 30). Bei LTC-Verlust wird der letzte Timecode gehalten (LTC NOT LOCKED). Die großen WARN-Meldungen (waiting for LTC lock, Clock not LTC synchronized, LTC reader not connected) sind **standardmäßig aus**, damit Scrubben/Videoschnitt WARN nicht flutet; **Show LTC unlock warning** einschalten, wenn sie gewünscht sind. Die Lock-LED zeigt immer LTC LOCK / LTC NOT LOCKED. NTP-Check vergleicht weiterhin die OS-Uhr mit dem NTP-Server, nicht den Timecode.

LBE-1110 Serial: USB-CDC-virtueller Seriellport eines [Leo-Bodnar-LBE-1110](#); keine Treiber. Auto wählt ein Leo-Bodnar-Gerät (USB-VID 0x1DD2) oder einen LBE-1110-/CDC-Port.

Audio Input: Dekodiert SMPTE-LTC (Biphase-Mark) von einem **lokalen** PortAudio-Capture-Gerät, unabhängig von den Audio Meters (kein Livewire/AES67). Kanal Links oder Rechts (Standard Links). Auf manchen Hosts kann dasselbe Gerät nicht zweimal geöffnet werden – wenn die Meters diesen Eingang schon nutzen, für LTC ein anderes Gerät wählen.

Die Lock-LED unten rechts ist grün, wenn die gewählte Quelle eingerastet ist, sonst rot. Daneben: PTP LOCK / PTP NOT LOCKED / PTP ERROR, NTP LOCK / NTP NOT LOCKED, LTC LOCK / LTC NOT LOCKED oder LOCAL.

Empfehlung: Einen lokalen NTP-Server im Studio-Netzwerk verwenden, da `pool.ntp.org` zeitweise unzuverlässig sein kann.

5.9 Audio Meters

Stereo-Pegelanzeige am linken Bildschirmrand: L/R (Sample-Peak, True Peak oder BBC PPM), ein einzelner Programme-LUFS-Balken (EBU R128) oder beides. L/R füllen mit RMS und legen den aktuellen Peak darüber. Programme-LUFS ist ein Balken (Momentary M, Short-term-S-Tick). Integrated I und LRA sind standardmäßig aus und starten mit LUFSI: START (Web-UI, MQTT-Home-Assistant-Schalter, Companion, OSC, UDP/HTTP oder Rechtsklick / Lang-Klick). LUFSI: RESET startet eine laufende Session neu; ist sie gestoppt, verschwinden I und LRA im Meter. Rechtsklick oder Lang-Klick auf den Main Screen → **Start I+LRA, Stop I+LRA** oder **Reset I+LRA**. Konfiguration unter **Settings** → **Audio Meters**.

Bestehende Configs mit Audio/unit=lufs (ohne layout) werden auf Layout lufs und L/R-Unit dbtp gemappt.

Einstellung	Schlüssel	Standard	Beschreibung
Enable Audio Meters	Audio/enabled	true	Meter-Spalte anzeigen
Audio Source	Audio/source	device	device, livewire oder aes67
Audio Input	Audio/input_device	(leer)	PortAudio-Gerätename (bei Source = Local Input)
Livewire Channel	Audio/livewire_channel	1	Livewire-Kanal 1–32767 (auch über Livewire Source)
AoIP Interface	Audio/livewire_iface	(leer)	IPv4 für Livewire/AES67-IGMP-Join; leer = Default
AES67 Stream ID	Audio/aes67_id	(leer)	SDP-Origin-Hash des gewählten Streams
AES67 Address	Audio/aes67_addr	(leer)	Multicast-Adresse für RTP-Capture
AES67 Port	Audio/aes67_port	5004	RTP-UDP-Port
AES67 Name	Audio/aes67_name	(leer)	Anzeigenname aus SDP s=
AES67 Codec	Audio/aes67_codec	L24	L16 oder L24
AES67 Sample Rate	Audio/aes67_rate	48000	44100, 48000 oder 96000
AES67 Channels	Audio/aes67_channels	2	Kanalzahl im Stream (Meter nutzt die ersten zwei)
AES67 Pasted SDP	Audio/aes67_manual	false	true, wenn der Stream per Paste SDP kam
Meter Layout	Audio/layout	both	lr, lufs oder both
Display Unit	Audio/unit	dbtp	L/R-Einheit: dbfs, dbtp, bbc_ppm (PPM nur bei lr)
Display Style	Audio/display_style	bargraph	solid oder bargraph

Einstellung	Schlüssel	Standard	Beschreibung
Meter Width	Audio/meter_width	115	Gesamtbreite in Pixel (53–201); extra Breite verdickt sichtbare Balken
LUFS Reference Preset	Audio/lufs_reference_preset	ebu_r128	ebu_r128, atsc_a85, aes_16, aes_18, custom
LUFS Reference	Audio/lufs_reference	–23.0	Zielpegel in LUFS (Peg auf der Skala)
Peak Hold	Audio/peak_hold	true	Peak-Marke halten
Peak Hold Seconds	Audio/peak_hold_seconds	1.5	Haltezeit der Peak-Marke
TooLoud	Audio/tooloud	true	Aktion bei True-Peak-Überschreitung
TooLoud Text	Audio/tooloudtext	T00 LOUD	Warntext
TooLoud Threshold	Audio/tooloud_threshold_dbtp	–1.0	Schwellwert in dBTP
TooLoud Action	Audio/tooloud_action	warning	warning oder led
TooLoud LED	Audio/tooloud_led	1	LED 1–4 bei Action = LED
Enable Silence Detection	Audio/silence	false	Aktuelle Audioquelle auf Stille überwachen
Show WARN in OAS	Audio/silence_warn	true	Silence-Alarm als WARN auf dem Bildschirm
Trigger when Device/Stream is absent	Audio/silence_on_absent	true	Fehlendes Device/Stream als Stille zählen
Silence Message	Audio/silence_text	SILENCE	WARN-Text (nur wenn Show WARN an ist)
Silence Threshold	Audio/silence_threshold_dbfs	–50.0	Sample-Peak-Schwellwert in dBFS (–90...0)
Max. Silence duration	Audio/silence_duration_s	10.0	Sekunden unter dem Schwellwert bis zum Alarm
Recovery time	Audio/silence_recovery_s	2.0	Sekunden über dem Schwellwert bis zum Löschen
HTTP GET URL	Audio/silence_http_url	(leer)	Optionale URL, einmalig beim Trigger aufgerufen

Silence Detection nutzt **dieselbe Audioquelle** wie die Meter (Local Input, Livewire oder AES67). Der Schwellwert ist immer **Sample-Peak dBFS**, unabhängig von der Meter-Anzeige. Capture läuft weiter, wenn Silence Detection an ist, auch wenn die Meter ausgeblendet sind.

Der Alarm rastet, wenn der Pegel die Duration lang unter dem Schwellwert bleibt, und löst nach der Recovery-Zeit über dem Schwellwert wieder. **Show WARN in OAS** ist unabhängig vom API/MQTT-Boolean: der Boolean wird aktualisiert, solange die Detection aktiv ist. Ein optionales HTTP GET wird nur bei der steigenden Flanke (false → true) ausgelöst.

Device/Stream absent: Wenn die Option an ist (Default), gilt eine Quelle, die nicht startet (kein Device, AES67 mit **None**, Startfehler), nach derselben Duration als Stille. Wenn die Option aus ist, zählen nur echte Pegel-Callbacks; Capture-Stop löscht einen hängenden Alarm. Paketverlust bei **laufendem** Stream (RTP-Timeout / Silence-Injection) zählt immer als Stille.

Das Bildschirm-WARN nutzt Priorität **2** (hoch) und liegt damit über TooLoud (Priorität 1). Das Flag `silence` in API/MQTT ist unabhängig von `texts.warn` und von `warning/active`.

Livewire: Der Rechner muss im AoIP-/Livewire-VLAN liegen (IGMP/Multicast). Solange die Settings (Desktop-Dialog oder WebUI-Overlay) offen sind, erscheinen announced Quellen in **Livewire Source** (239.192.255.3 UDP **4001**, nur Standard-Stereo-Streams). Quelle wählen oder Kanalnummer eintippen. Kanal *N* entspricht Multicast 239.192.0.0 + *N* auf UDP-Port **5004** (48 kHz / 24Bit Stereo RTP). Nach Apply läuft der Empfang mit gespeichertem Kanal weiter, auch wenn die Ads verstummen.

AES67: SAP-Discovery lauscht auf 239.255.255.255 und RFC 2974 224.2.127.254 UDP **9875**, solange der Einstellungsdialog oder das WebUI-Overlay offen ist. Die Stream-Liste aktualisiert sich live **nur wenn Streams dazukommen oder verschwinden** (kein Flackern bei unveränderter Liste). Die Combo hat immer **None** (kein Stream). SAP-Einträge verschwinden, wenn sie nicht mehr announced werden (Deletion oder Timeout). **Paste SDP**, falls ein Gerät nicht per SAP announced — diese Einträge bleiben in der Liste und sind mit **pasted SDP** gekennzeichnet. Unterstützt: L16/L24 bei 44,1/48/96 kHz, 1–64 Kanäle (Meter zeigt Kanal 1–2). Dante-AES67-SAP-Streams erscheinen wie andere (a=keywords:Dante nur als Label). Kein PTP und kein Playout — nur Metering. Nach Apply läuft der Empfang mit gespeicherter Adresse/Port weiter, auch wenn SAP gerade schweigt. Apply mit **None** beendet den AES67-Empfang und setzt das Meter zurück.

Lokaler Eingang: unter macOS Mikrofon-Berechtigung für OnAirScreen.

5.10 GPIO / MIDI

GPIO legt Kontakte, Livewire-Source-Pins und MIDI-Befehle auf dieselben LED- und AIR-Kommandos wie die Netzwerk-API. Einstellen unter **Settings** → **GPIO**. Innere Reiter (oben): **Raspberry Pi**, **Axia Livewire**, **MIDI** und **Logic**. Pi, Livewire und MIDI können gleichzeitig aktiv sein. Logic verknüpft On/Off-Pegel aus allen dreien.

Raspberry Pi

Raspberry-Pi-GPIO-Eingänge legen Mixer-GPI-Kontakte. Nur auf dem Raspberry Pi verfügbar; auf anderen Plattformen zeigt dieser Unter-Tab „not available“.

Pi-GPIO ist **3,3 V**. Mixer-GPI ist oft 5–24 V oder Open Collector — **immer ein PC817-Isolationsmodul verwenden**. Potenzialfreie Relaiskontakte nach GND dürfen direkt angeschlossen werden (interner Pull-up, Invert an).

Einstellung	Schlüssel	Standard	Beschreibung
Enable GPIO	GPIO/enabled	false	Konfigurierte BCM-Pins überwachen
Debounce	GPIO/debounce_ms	50	Prellen kürzer als dieses Intervall ignorieren (ms)
GPIn Enable	GPIO/gpiN_enabled	GPiI-2 an, 3-8 aus	Diesen Eingang nutzen
GPIn Name	GPIO/gpiN_name	(leer)	Kurzes Label für Logic-Regeln, z. B. MIC1
GPIn BCM pin	GPIO/gpiN_pin	17, 27, 5, 6, 12, 13, 16, 22	Nur sichere BCM-Pins
GPIn Invert	GPIO/gpiN_invert	true	An: Kontakt nach GND ist aktiv (Pull-up)
GPIn Mode	GPIO/gpiN_mode	level	level, rising, falling, both
GPIn Actions	GPIO/gpiN_action	LED1 / AIR3	Eine oder mehrere von LED1-4, AIR1-4, AIR3/AIR4 Reset (kommagetrennt)
GPIn Command	GPIO/gpiN_command	(leer)	Zusätzliche API-Kommandos, getrennt durch Semikolon

Level (typisch Tally/Fader): geschlossen sendet LED1:ON / AIR3:ON für jede gewählte Action, offen sendet OFF. Custom-Kommandos nur bei geschlossenem Kontakt. **Rising / Falling / Both** senden TOGGLE (oder Reset / die Custom-Kommandos) auf dieser Flanke. Ein altes CUSTOM im Action-Feld bedeutet weiterhin „nur Custom-Kommandos“.

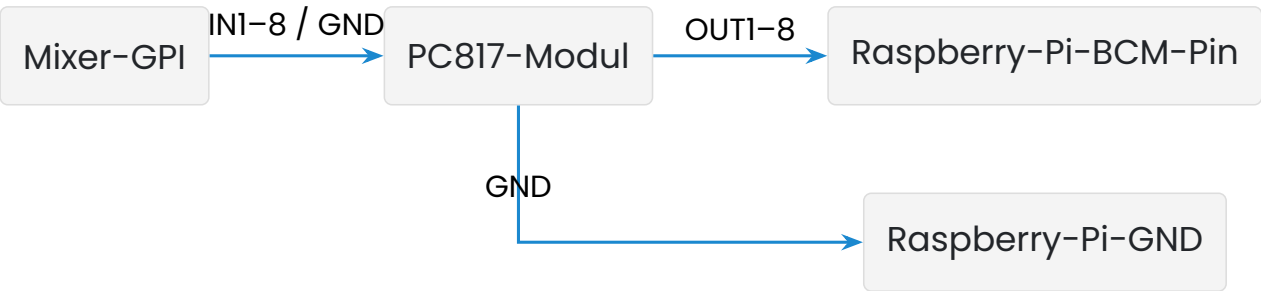
Werks-Mapping für zwei Mixer-GPI-Leitungen: GPiI BCM 17 → LED1 (ON AIR), GPi2 BCM 27 → AIR3 (Radio-Timer). GPIO aktivieren und Kontakte über ein PC817-Modul (oder trockene Relais nach GND) anschließen.

Sichere BCM-Pins: 5, 6, 12, 13, 16, 17, 22, 23, 24, 25, 26, 27 (nicht I2C 2/3 oder UART 14/15).

Verdrahtung

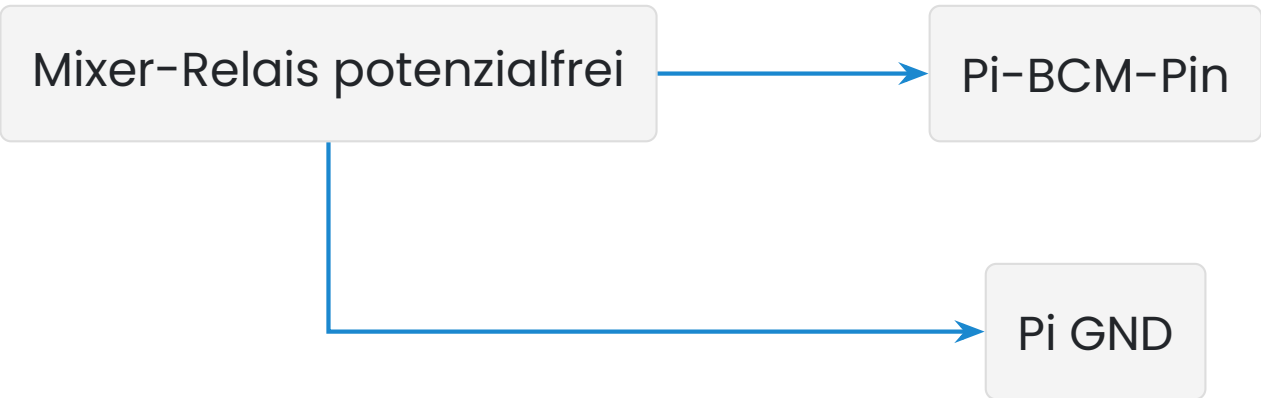
Der 40-polige GPIO-Header ist derselbe auf Raspberry Pi **3, 4, 400, 5, 500, Zero 2 W** und dem **CM4/CM5**-IO-Board-Header. In den Settings gelten **BCM**-Nummern, nicht die physischen Stiftnummern.

PC817-Modul (empfohlen bei Mixer-GPI 5-24 V oder Open Collector):



Jeder Mixer-GPI besteht aus zwei Adern: IN1–IN8 dieses Kanals und das GND der Eingangsseite. Auf der Pi-Seite denselben Kanal OUT1–OUT8 an den BCM-Pin und das GND der Ausgangsseite an Pi-GND (GPI1 → IN1/OUT1, GPI2 → IN2/OUT2, ...). **Invert** anlassen (aktiv = Pin nach GND gezogen). Beispiel: [Hailege 8-Kanal-PC817-Isolationsmodul](#) (Amazon-Affiliate-Link).

Nur trockener Relaiskontakt (bereits isoliert, keine Spannung am Mixer-GPI):



Keine 5 V oder 12/24 V direkt auf einen GPIO-Pin legen.

Werks-Defaults am Header (Platine mit USB/Ethernet nach unten; Stift 1 ist bei den meisten Boards die 3,3-V-Ecke am SD-/Power-Ende):

OAS-Eingang	BCM	Header-Stift	Typische Nutzung
GPI1	17	11	ON AIR (LED1)
GPI2	27	13	Radio-Timer (AIR3)
Masse	—	6, 9, 14, 20, 25, 30, 34, 39	gemeinsames GND

Sichere BCM-Pins und zugehörige Header-Stifte (diese Werte unter **BCM pin**):

BCM	Header	BCM	Header	BCM	Header
5	29	12	32	22	15
6	31	13	33	23	16
16	36	17	11	24	18
26	37	27	13	25	22

Pinout-Referenzen (40-poliger Header, BCM-Nummerierung):

- [Raspberry Pi GPIO and 40-pin header](#) (offiziell)
- [pinout.xyz](#) (interaktiv; gleicher Header auf Pi 3 / 4 / 5 / Zero 2 W)
- Pi 5 Überblick: [Raspberry Pi 5](#)
- Compute-Module-IO-Boards: [Compute Module](#)

Axia Livewire

Axia-Livewire-GPIO folgt Source-Pins (Operator-Mic, Guest-Mic, Line, ...) und legt sie auf dieselben LED- und AIR-Kommandos. OnAirScreen **liest** GPIO nur (es schreibt kein GPI zurück zur Konsole). Verfügbar auf allen Plattformen. Unter-Tab **Axia Livewire**.

Zwei Modi:

- **Multicast** (Standard): CMsg2 auf 239.192.255.4. Die Mapping-Spalte **Channel** ist der Livewire-**Audiokanal** dieser Quelle. Der Rechner muss im AoIP-/Livewire-VLAN liegen (IGMP/Multicast); dasselbe Interface wie bei den Livewire-Audio-Metern wählen (leer = Default-Route).
- **LWRP (TCP)**: Verbindung zu einem Gerät (xNode GPIO, iPort, Konsole, ...) unter **IP:93**. Die Mapping-Spalte heißt **Port** – GPIO-Port am Gerät, nicht Livewire-Audiokanal. Kein IGMP-Join; der Host muss im Livewire-/AoIP-Netz erreichbar sein.

Typisches Mikrofon-ON-AIR-Profil: **GPO Pin 1** ist die ON-Lampe (latchend, solange der Kanal ON ist; bei TALK/MUTE oft aus). GPI 1 wäre der ON-Befehl zur Konsole – OAS sendet das nicht.

Einstellung	Schlüssel	Standard	Beschreibung
Enable Livewire GPIO	GPIO/lw_enabled	false	Livewire-GPIO folgen (Multicast oder LWRP)
Mode	GPIO/lw_transport	multicast	multicast oder lwrp
Host	GPIO/lw_host	(leer)	Geräte-IPv4 (LWRP)
Port	GPIO/lw_port	93	LWRP-TCP-Port
Password	GPIO/lw_password	(leer)	Optionales LWRP-LOGIN-Passwort
AoIP-Interface	Audio/livewire_iface	(leer)	IPv4-Interface für IGMP (Multicast-Modus; wie Audio-Meter)
Debounce	GPIO/lw_debounce_ms	50	Prellen kürzer als dieses Intervall ignorieren (ms)
GPIIn Enable	GPIO/lw_gpiN_enabled	Zeilen 1–2 an	Diese Mapping-Zeile nutzen
GPIIn Name	GPIO/lw_gpiN_name	(leer)	Kurzes Label für Logic-Regeln, z. B. MIC1
GPIIn Channel/Port	GPIO/lw_gpiN_channel	1, 2, dann Zeile <i>n</i>	Livewire-Kanal (Multicast) oder GPIO-Port am Gerät (LWRP), 1–32767
GPIIn Direction	GPIO/lw_gpiN_dir	gpo	gpo (von der Konsole) oder gpi (zur Konsole)
GPIIn Pin	GPIO/lw_gpiN_pin	1	Pin 1–5 auf diesem Kanal/Port

Einstellung	Schlüssel	Standard	Beschreibung
GPIIn Invert	GPI0/lw_gpiN_invert	false	Aus: Livewire Low (L/I) ist aktiv
GPIIn Mode	GPI0/lw_gpiN_mode	level	level, rising, falling, both
GPIIn Actions	GPI0/lw_gpiN_action	LED1 / AIR3	Eine oder mehrere von LED1–4, AIR1–4, AIR3/AIR4 Reset (kommagetrennt)
GPIIn Command	GPI0/lw_gpiN_command	(leer)	Zusätzliche API-Kommandos, getrennt durch Semikolon

Werks-Mapping (Master **Enable Livewire GPIO** bleibt aus): Zeile 1 Kanal/Port 1 GPO Pin 1 → LED1, Zeile 2 Kanal/Port 2 GPO Pin 1 → AIR3.

Multicast ist 239.192.255.4 UDP **2055** (GPIO-Node → Konsole) und **2060** (Konsole → Node). OAS joined beide. Nach dem Join fordert OAS den aktuellen Pin-Zustand an, damit ein Mikrofon, das beim Start schon ON ist, die LED setzt.

LWRP: OAS öffnet TCP 93, sendet LOGIN (mit Passwort, falls gesetzt), ADD GPI, ADD GPO und fragt dann jeden beobachteten Port ab, damit ein Mikrofon, das beim Start schon ON ist, die LED setzt. Bei Verbindungsabbruch wird neu verbunden.

Level (typische ON-Lampe): GPO 1 low sendet LED1: 0N, high sendet 0FF. Invert bleibt **aus**, außer die Konsolen-Polarität ist umgekehrt.

MIDI

MIDI liest ein Eingangs-Interface (Settings → GPIO → MIDI) und legt Note- oder CC-Befehle auf dieselben LED- und AIR-Kommandos. **Learn** wartet bis zu 10 Sekunden auf die nächste Note oder den nächsten CC und füllt Kanal, Typ und Nummer. Clock, Active Sensing und SysEx werden ignoriert.

Einstellung	Schlüssel	Standard	Beschreibung
Enable MIDI	GPI0/midi_enabled	false	Am gewählten Interface mithören
Interface	GPI0/midi_port	(leer)	Name des Eingangs. Ein fehlendes Gerät bleibt als „not connected“ stehen
Debounce	GPI0/midi_debounce_ms	20	Oneshot und Toggle fassen einen Burst kürzer als dieses Intervall als einen Druck zusammen (ms)
MIDIIn Enable	GPI0/midiIn_enabled	false	Diese Zeile nutzen
MIDIIn Name	GPI0/midiIn_name	(leer)	Kurzes Label für Logic, z. B. PAD
MIDIIn Channel	GPI0/midiIn_channel	1	MIDI-Kanal 1–16
MIDIIn Type	GPI0/midiIn_kind	note	note oder cc

Einstellung	Schlüssel	Standard	Beschreibung
MIDIn Number	GPI0/midiN_number	0	Noten- oder CC-Nummer, 0–127
MIDIn Threshold	GPI0/midiN_threshold	64	CC ab diesem Wert ist aktiv
MIDIn Behavior	GPI0/midiN_behavior	follow	follow (On/Off), oneshot oder toggle
MIDIn Invert	GPI0/midiN_invert	false	On/Off tauscht den Haltepegel. Oneshot sendet OFF statt ON
MIDIn Actions	GPI0/midiN_action	(leer)	LED1–4, AIR1–4, AIR3/AIR4 Reset
MIDIn Command	GPI0/midiN_command	(leer)	Zusätzliche API-Kommandos, semikolongetrennt

On/Off hält den Pegel: Note On (Velocity > 0) oder ein CC ab der Schwelle sendet LED1:ON, Note Off oder ein CC unter der Schwelle sendet OFF. **Oneshot** sendet einmal pro Druck LED1:ON, mit Invert LED1:OFF. **Toggle** sendet einmal pro Druck LED1:TOGGLE. Reset und Custom feuern bei beiden einmal, auch bei invertiertem Oneshot. Beide lösen erneut aus, auch wenn der Controller nie ein Note Off schickt. Ein Regler, der oberhalb der Schwelle weiter CCs sendet, bleibt ein Burst. Zwei eingeschaltete Zeilen mit demselben Kanal, Typ und derselben Nummer feuern beide, und der Tab zeigt eine Warnung. Solange die Einstellungen offen sind, blinkt eine eingeschaltete Zeile, wenn eine Note oder ein CC mit ihrem Kanal, Typ und ihrer Nummer ankommt.

In Logic stehen nur On/Off-Zeilen. Oneshot und Toggle fehlen im Logic-Menü, auch wenn die Zeile eingeschaltet ist und einen Namen hat, weil sie keinen Haltepegel haben. Jede On/Off-Zeile steht in der Liste, auch wenn Enable aus ist. Dieser ausgeschaltete Eingang zählt in der Regel als aus. On/Off-Zeilen ohne Namen heißen MIDI1 ... MIDI8. Mit Namen steht dort PAD (MIDI1). Damit eine MIDI-Zeile in Logic auftaucht, Behavior auf On/Off stellen.

Logic

Logic verknüpft Raspberry-Pi-, Livewire- und MIDI-On/Off-Eingänge. Jede Regel hat einen Operator, einen oder mehrere Eingänge und einen oder mehrere Ausgänge. Die Ausgänge folgen dem Gesampegel: ein Kommando geht nur raus, wenn dieses Ergebnis umschlägt. Das Eingangsmenü enthält jeden Pi-Eingang, jeden Livewire-Eingang und jede MIDI-Zeile mit Behavior On/Off. MIDI-Oneshot und MIDI-Toggle fehlen. Eine MIDI-On/Off-Zeile bleibt im Menü, wenn Enable aus ist.

Operator	Ausgänge an	Ausgänge aus
OR	irgendein gewählter Eingang ist aktiv	alle gewählten Eingänge sind inaktiv
AND	alle gewählten Eingänge sind aktiv	irgendein gewählter Eingang ist inaktiv

Beispiel: Pi GPI1 MIC1 und Pi GPI2 MIC2 benennen. Regel OR mit diesen beiden Eingängen

und den Ausgängen LED1 und AIR1 bleibt an, solange ein Mikrofon offen ist, und geht erst aus, wenn beide zu sind.

Ausgeschaltete, fehlende oder noch nicht gesehene Eingänge zählen als aus. Eine leere Eingangsliste bleibt aus. Resets und Custom-Kommandos feuern, wenn die Regel wahr wird, nicht wenn sie falsch wird. Direkte Zeilen-Aktionen (einschließlich Flanken) laufen parallel weiter. Steuert dieselbe LED oder derselbe AIR-Timer schon eine Zeile oder eine andere Regel, zeigt die Logic-Zeile eine Warnung.

Einstellung	Schlüssel	Standard	Beschreibung
Rule <i>n</i> Enable	GPI0/logicN_ enabled	false	Diese Regel nutzen
Rule <i>n</i> Operator	GPI0/logicN_op	or	or oder and
Rule <i>n</i> Inputs	GPI0/logicN_inputs	(leer)	pi:1, lw:2, midi:3, kommagetrennt
Rule <i>n</i> Outputs	GPI0/logicN_action	(leer)	LED1-4, AIR1-4, AIR3/AIR4 Reset
Rule <i>n</i> Command	GPI0/logicN_ command	(leer)	Zusätzliche API-Kommandos, wenn die Regel wahr wird

Acht Regeln. Ein Mikrofon, das beim Start schon offen ist, schaltet eine passende Regel an, sobald der Pegel dieses Eingangs bekannt ist.

5.11 About

Element	Beschreibung
Version	Aktuelle OnAirScreen-Version
Distribution	OpenSource oder kommerzielle Distribution
Settings Path	Pfad zur Konfigurationsdatei auf diesem System
Log Folder	Ordner mit onairscreen.log und Crash-Reports
Open log folder	Öffnet diesen Ordner im Dateimanager
Loglevel	DEBUG, INFO, WARNING, ERROR, CRITICAL, NONE
Enable Reset	Checkbox zum Freischalten des Reset-Buttons
Reset all settings	Setzt alle Einstellungen auf Standardwerte zurück (unwiderruflich)
Reboot	Host-Betriebssystem neu starten (mit Nachfrage)
Shutdown	Host-Betriebssystem herunterfahren (mit Nachfrage)

6. Funktionen im Detail

6.1 Status-LEDs

Jede LED kann einzeln ein- und ausgeschaltet werden (Tasten 1–4, Linksklick, oder Fernsteuerung). Im aktiven Zustand werden konfigurierte Vorder- und Hintergrundfarben verwendet; im inaktiven Zustand die gemeinsamen Inaktiv-Farben.

Blinkmodi:

- **Autoflash:** Blinkt dauerhaft im 500-ms-Takt, solange die LED eingeschaltet ist
- **20sec flash:** Blinkt 20 Sekunden, schaltet sich dann automatisch aus

6.2 AIR-Timer

Alle AIR-Timer zählen die verstrichene Zeit im Format **M:SS** (z. B. 3:45). AIR3 zeigt ▲ (Hochzählen) oder ▼ (Runterzählen) unter dem Timer-Icon.

AIR1 (Mikrofon) und AIR2 (Telefon)

- Einfache Stoppuhr: Start/Stop, Sekunden werden bei Start auf 0 zurückgesetzt
- Steuerung: M// (AIR1), P/* (AIR2), oder Linksklick auf den Timer

AIR3 (Radio-Timer)

Der vielseitigste Timer mit drei Betriebsarten:

1. **Hochzählen (Count-Up):** Standardmodus, zählt von 0:00 aufwärts
2. **Runterzählen (Count-Down):** Wird per AIR3TIME: seconds oder Timer-Dialog gesetzt
3. **Top-of-Hour (TOH):** Countdown bis zur nächsten vollen Stunde (Format MM:SS, z. B. 22:38)

Start/Stop: Leertaste, , / ., oder Linksklick auf den Timer.

Top-of-Hour-Verhalten:

- Erster Aufruf: Berechnet verbleibende Zeit bis :00, startet Countdown, Beschriftung ist der konfigurierte TOTH-Timer-Text (Standard TOTH Timer)
- Zweiter Aufruf (oder OFF/TOGGLE während aktiv): Stoppt und stellt die konfigurierte AIR3-Beschriftung wieder her
- Synchronisiert mit der Anzeigeuhr, stoppt automatisch bei Stundenwechsel und stellt die konfigurierte Beschriftung wieder her
- API-Status: "topOfHour": true und "text" mit dem konfigurierten TOTH-Timer-Text in air[3] wenn aktiv

Timer-Eingabedialog (Enter):

Eingabe	Bedeutung
2:10, 2, 10 oder 2.10	2 Minuten 10 Sekunden (Count-Down)
30	30 Sekunden (Count-Down)
0	Count-Up-Modus

AIR4 (Stream-Timer)

- Wie AIR3, aber ohne Top-of-Hour-Funktion
- Steuerung: S oder Linksklick (Start/Stopp), Alt+S (Reset)

6.3 Textzeilen NOW, NEXT, WARN

Zeile	API-Befehl	Beschreibung
NOW	NOW: TEXT	Erste Fußzeile (aktueller Titel, IP-Adressen, ...)
NEXT	NEXT: TEXT	Zweite Fußzeile (nächster Titel, ...)
WARN	WARN: TEXT	Warnmeldung mit rotem Warnmodus

Maximale Textlänge: 500 Zeichen (Eingaben werden automatisch bereinigt und gekürzt).

Warnungssystem mit Prioritäten

Priorität	Bedeutung	API-Format
-1	NTP-Warnung (automatisch)	<i>(intern)</i>
0	Normal / Legacy	WARN: TEXT
1	Medium	WARN: 1: TEXT
2	High (höchste)	WARN: 2: TEXT

Anzeigeregeln: Die Warnung mit der **höchsten Priorität** wird angezeigt. NTP-Warnungen (-1) erscheinen nur, wenn keine andere Warnung aktiv ist. Bei aktiver Warnung werden NOW und NEXT ausgeblendet.

Warnung löschen:

Methode	Befehl
Priorität 0	WARN: <i>(leerer Text)</i>
Priorität 1	WARN: 1: <i>(leerer Text nach Doppelpunkt)</i>
Priorität 2	WARN: 2: <i>(leerer Text nach Doppelpunkt)</i>
Web-UI	X-Button neben der Warnung

6.4 IP-Adressen anzeigen

Taste I (oder automatisch beim Start) zeigt alle lokalen IPv4-Adressen in **NOW** und IPv6-Adressen in **NEXT** für 10 Sekunden an.

Wenn **Replace IPs after 10s** aktiviert ist, wird die NOW-Zeile danach durch den konfigurierten Ersatztext (`replacenowtext`) ersetzt.

6.5 Uhr

- **Digital:** LED-Style-Ziffernanzeige mit konfigurierbaren Farben
- **Analog:** Klassisches Zifferblatt ohne Zahlen
- **Analog Numbers:** Klassisches Analog-Face mit 1–12
- **Analog Studio:** Weißes Zifferblatt, schwarze Ticks, Zahlen 1–12
- **Analog Railway:** Weißes Bahnhofszifferblatt ohne Zahlen, spitze schwarze Zeiger, kontinuierlicher roter Sekundenzeiger mit Ring
- **Analog 24h smooth:** Cremefarbenes Zifferblatt, schwarze 1–12 und rote 13–24, kontinuierlicher Sekundenzeiger
- **Analog 24h ticking:** Dasselbe 24h-Zifferblatt mit tickendem Sekundenzeiger



Abbildung 2: Digitales Zifferblatt



Abbildung 3: Analoges Zifferblatt

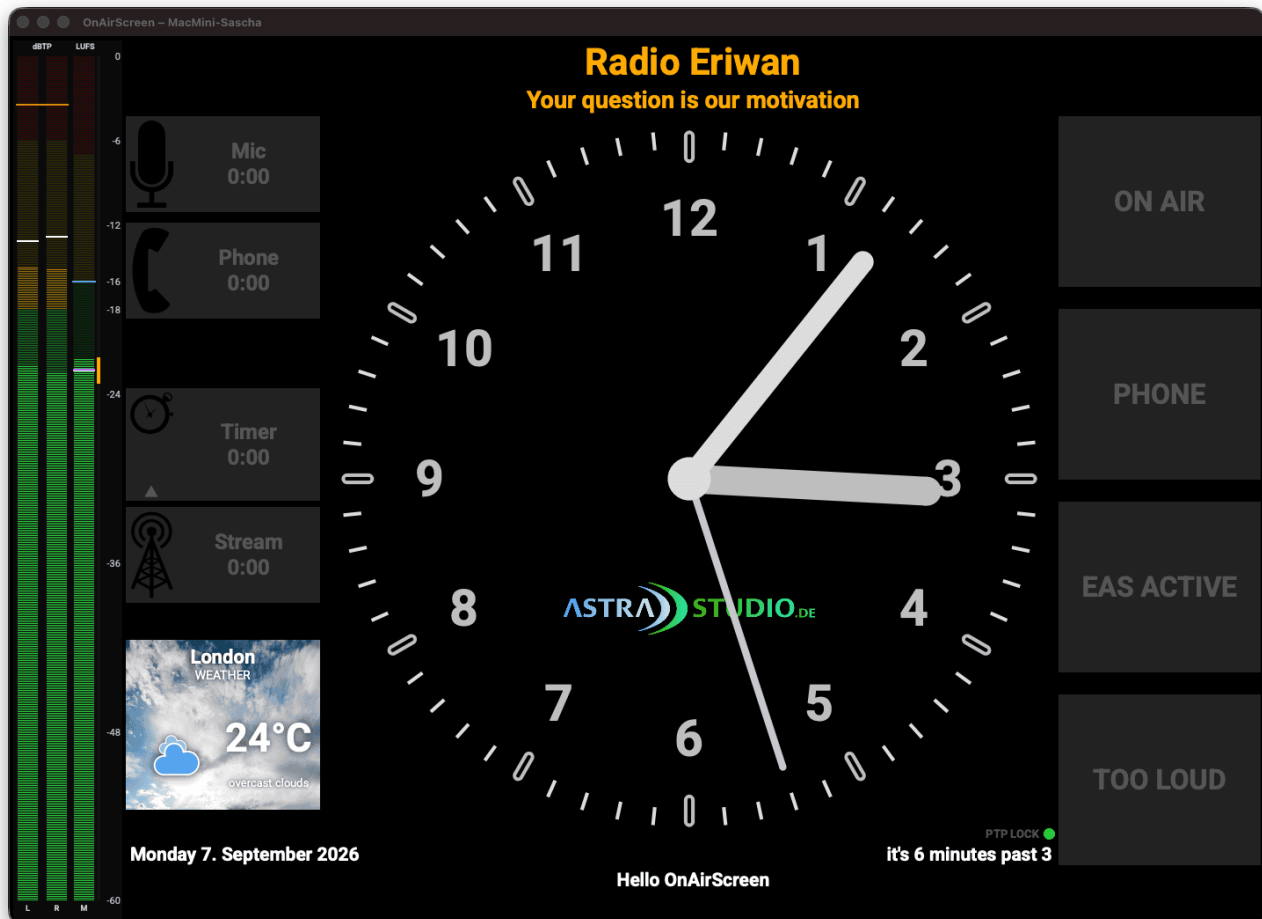


Abbildung 4: Analog Numbers



Abbildung 5: Analog Studio



Abbildung 6: Analog Railway

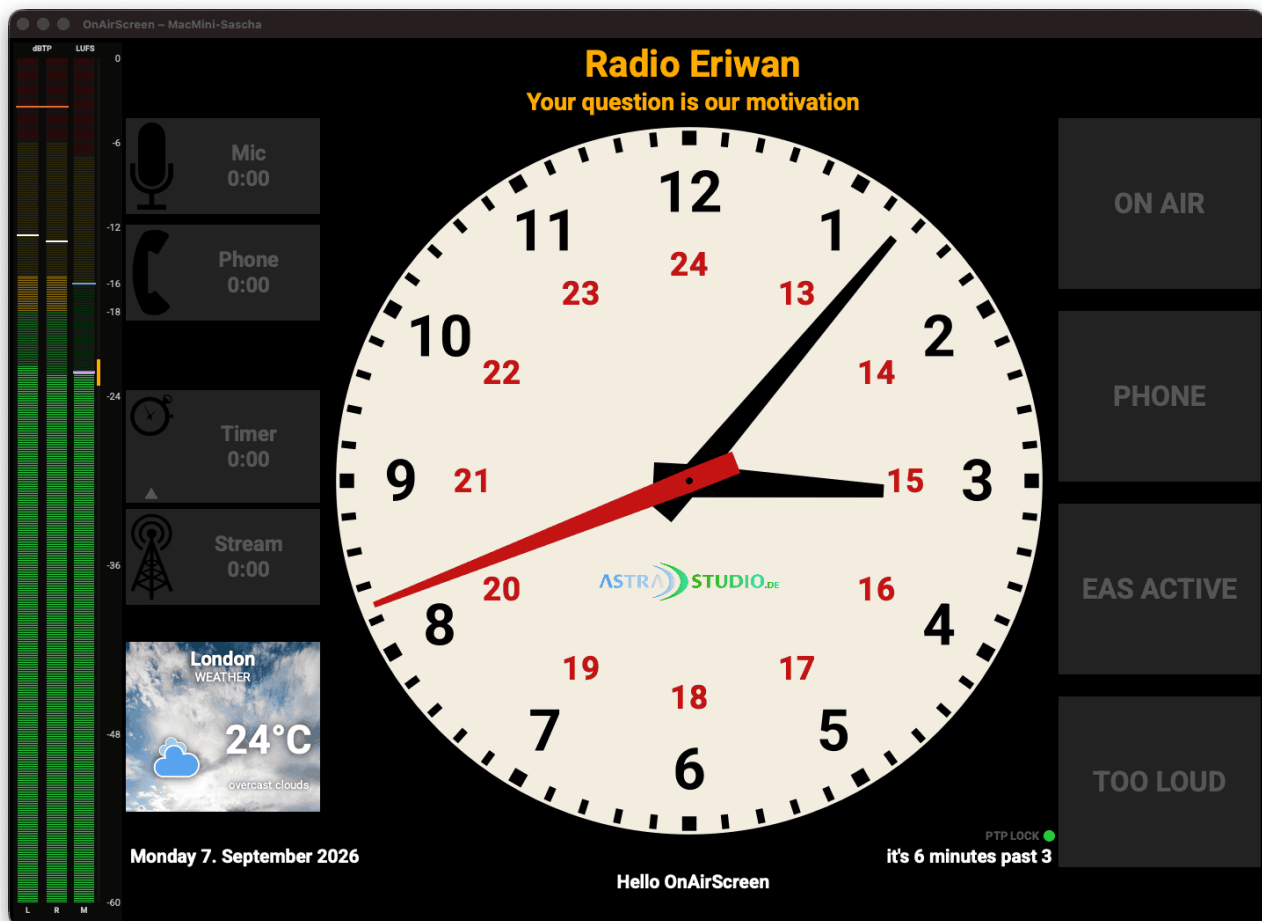


Abbildung 7: Analog 24h (kontinuierlicher oder tickender Sekundenzeiger)

- **Textuhr:** Sprachliche Zeitanzeige (z. B. „it's a quarter past three“)
- **Zeitquelle:** Local, NTP, PTPv2 oder LTC — siehe 5.8 Time Source; die OS-Uhr wird nie gesetzt
- **Lock-LED** unten rechts: grün = eingerastet, rot = nicht eingerastet (PTP LOCK / NTP LOCK / LTC LOCK oder LOCAL)
- **Wetter-Widget:** Optional rechts neben der Uhr (OpenWeatherMap)

6.6 Systembefehle

Befehl	Funktion
CMD:REBOOT	Betriebssystem neu starten
CMD:SHUTDOWN	Betriebssystem herunterfahren
CMD:QUIT	OnAirScreen beenden

Reboot und Shutdown gibt es auch unter **Settings** → **About** und in der Web-UI unter **System**. Beide fragen nach. HTTP CMD:REBOOT / CMD:SHUTDOWN verlangen die Web-Settings-PIN, wenn eine gesetzt ist (X-Settings-Token). UDP, MQTT, OSC, GPIO und SNMP bleiben ohne Auth. CMD:QUIT bleibt nur per API.

6.7 GPIO-Eingänge

Mixer-GPI (Kontakt) kann LEDs und AIR-Timer über **Raspberry-Pi-GPIO** ansteuern. Axia-Konsolen können dieselben LED- und AIR-Aktionen über **Livewire-GPIO** ansteuern (Multicast-Source-Kanal oder LWRP TCP 93, z. B. Mic-ON-Lampe). Ein MIDI-Controller kann sie über Note- und CC-Befehle an einem Eingang ansteuern. Alle drei können gleichzeitig laufen. Pi-Verdrahtung, Livewire-Kanal/Port-Mapping, MIDI-Learn und das AoIP-Interface stehen unter 5.10 GPIO. Pi-GPIO ist lokale Hardware; Livewire-GPIO ist Multicast oder eine TCP-Sitzung im AoIP-Netz; MIDI ist der gewählte Eingang.

7. Fernsteuerung und API

OnAirScreen unterstützt Fernsteuerung über UDP, HTTP, Web-UI, MQTT, OSC, SNMP, REST-API und Bitfocus Companion.

7.1 UDP (Port 3310)

```
# LED1 einschalten
echo "LED1:ON" > /dev/udp/127.0.0.1/3310

# NOW-Text setzen
echo "NOW:Aktueller Songtitel" > /dev/udp/127.0.0.1/3310

# Konfiguration ändern
echo "CONF:LED1:text=STUDIO LIVE" > /dev/udp/127.0.0.1/3310
echo "CONF:CONF:APPLY=TRUE" > /dev/udp/127.0.0.1/3310
```

7.2 HTTP (Port 8010)

```
curl "http://127.0.0.1:8010/?cmd=LED1:ON"
curl "http://127.0.0.1:8010/?cmd=NOW:Aktueller%20Song"
```

7.3 Web-UI

Browser öffnen: `http://<IP-Adresse>:8010/`

Funktionen der Web-UI:

- Echtzeit-Status für LEDs, AIR-Timer, Textfelder, Silence-Alarm und Loudness I+LRA (I- und LRA-Werte plus Laufstatus)
- Instanzname in Titel und Status (aus `/api/status instance`)
- WebSocket-Updates (mit HTTP-Polling-Fallback; nach Polling wird wieder WebSocket versucht)
- Dark Mode mit persistenter Theme-Einstellung
- LED- und Timer-Steuerung mit Toggle-Buttons
- Start / Stop / Reset für Programme-I + LRA (LUFISI)
- Top-of-Hour-Button für AIR3
- AIR3-Zeit setzen (m:ss, m,ss oder Sekunden)
- AIR3 Count-up vs. Countdown in der Statuskachel
- Tasten 1-4 schalten LEDs (nicht während der Texteingabe)
- Texteingabe für NOW, NEXT, WARN (NOW/NEXT folgen dem Live-Status, solange nicht editiert wird)
- Warnungen mit Priorität und Lösch-Button
- Versions- und Distributionsinformationen
- Dauerhaftes Connection-Badge (Live / Polling / Offline) plus Fehler-Modal

- Zahnrad oben rechts: Settings-Overlay für alle editierbaren Einstellungen, optionale PIN, Apply sowie Presets laden/speichern
- Links im Overlay: dieselbe Nearby-Liste, **Add as replica** / **Use as source** und Replicas-/Sources-Raster wie in den Desktop-Settings
- Audio Meters im Overlay: Dropdowns **Livewire Source** und **AES67 Stream** (Live-Discovery solange das Overlay offen ist) plus Paste SDP

7.4 REST-API

Status abfragen:

curl http://127.0.0.1:8010/api/status

Antwort (vereinfacht):

```
{
  "leds": { "1": { "status": true, "text": "ON AIR", "autoflash": false } },
  "air": { "3": { "status": false, "seconds": 0, "text": "Timer", "topOfHour":
false, "countDown": false } },
  "texts": { "now": "Song", "next": "Next Song", "warn": "" },
  "warnings": [],
  "silence": false,
  "lufsIntegrated": false,
  "lufsI": null,
  "lra": null,
  "instance": "Studio-1",
  "version": "1.0.10",
  "distribution": "OpenSource",
  "link": { "enabled": false, "follow": null, "replicas": [], "peers": 0 }
}
```

Das Feld `silence` ist `true`, solange Silence Detection eingerastet ist, auch wenn das Bildschirm-WARN ausgeschaltet ist. `lufsIntegrated` ist `true`, solange eine I+LRA-Session läuft (LUFSI:START). `lufsI` ist die gated Integrated Loudness in LUFS (eine Nachkommastelle), `lra` die Loudness Range in LU; beide sind `null`, bis genug Audio gemessen wurde. `instance` ist der konfigurierte Instanzname. Bei AIR3 zeigt `topOfHour`, ob der Top-of-Hour-Countdown aktiv ist, und `countDown`, ob der Radio-Timer runterzählt.

Befehl senden:

curl "http://127.0.0.1:8010/api/command?cmd=LED1:ON"

Instanz-Links (ohne Web-Settings-PIN):

curl http://127.0.0.1:8010/api/link/peers

curl -X POST http://127.0.0.1:8010/api/link/follow -H 'Content-Type: application/json' \

-d '{"source_instance":"Studio-1","source_http_port":8010}'

curl -X POST http://127.0.0.1:8010/api/link/unfollow -H 'Content-Type: application/json' \

```
-d '{"source_instance":"Studio-1"}'
```

link in `/api/status` zeigt, ob dieser Screen ein Replica ist (follow) und an welche Instanzen er pusht (replicas).

Web-Settings (optionale PIN über X-Settings-Token nach POST `/api/settings/auth`):

```
curl http://127.0.0.1:8010/api/settings/auth
curl -X POST http://127.0.0.1:8010/api/settings/auth -H 'Content-Type: application/json' -d '{"pin":"1234"}'
curl http://127.0.0.1:8010/api/settings
curl "http://127.0.0.1:8010/api/settings/aoip?source=livewire&channel=1"
curl -X PUT http://127.0.0.1:8010/api/settings -H 'Content-Type: application/json' -d '{"config":{"General":{"slogan":"On air"}}}'
```

Geheimnisse (updatekey, MQTT-Passwort, OpenWeatherMap-API-Key) kommen im Klartext zurück, damit das Overlay sie anzeigen kann. Die PIN kommt als `__unchanged__` zurück (gespeichert wird nur der Hash). Diesen Sentinel senden, um die PIN zu behalten; – entfernt sie. Änderungen an UDP-/HTTP-Port werden gespeichert, greifen aber erst nach einem App-Neustart.

7.5 MQTT

Befehle senden (Topic: `{base_topic}/...`):

Topic	Payload	Funktion
<code>led{1-4}/set</code>	ON / OFF / TOGGLE	LED schalten
<code>air{1-4}/set</code>	ON / OFF / TOGGLE	Timer starten/stoppen
<code>air{3-4}/reset</code>	PRESS	Timer zurücksetzen
<code>air3/toh</code>	ON / OFF / TOGGLE	Top-of-Hour
<code>lufs/integrated/set</code>	ON / OFF / TOGGLE / RESET	I+LRA starten/stoppen/reset
<code>lufs/integrated/reset</code>	PRESS	I+LRA zurücksetzen
<code>text/now/set</code>	TEXT	NOW-Text setzen
<code>text/next/set</code>	TEXT	NEXT-Text setzen
<code>text/warn/set</code>	TEXT	WARN-Text setzen

Status-Topics (automatisch publiziert):

Topic	Payload
<code>led{1-4}/state</code>	ON / OFF
<code>air{1-4}/state</code>	ON / OFF
<code>air{1-4}/time</code>	Sekunden (Integer)
<code>air3/toh/state</code>	true / false
<code>text/{now,next,warn}/state</code>	Text
<code>warning/active</code>	true / false
<code>silence/active</code>	true / false
<code>lufs/integrated/state</code>	ON / OFF

Topic	Payload
lufs/i	I in LUFS ("" wenn unbekannt)
lufs/lra	LRA in LU ("" wenn unbekannt)
instance/state	Instanzname

Home Assistant Autodiscovery erstellt automatisch:

- LED-Schalter (LED1–4)
- AIR-Timer-Schalter (AIR1–4)
- AIR-Zeit-Sensoren (AIR1–4 Time)
- Reset-Buttons (AIR3/AIR4)
- Top-of-Hour-Button (AIR3)
- Text-Entitäten (NOW, NEXT, WARN)
- Binary Sensoren: Warning Active, Silence
- Sensor: Instance (Instanzname)
- Loudness-I+LRA-Schalter (Start setzt zurück; Stop friert die letzten Werte)
- Loudness-I+LRA-Reset-Taste (laufend neu starten; gestoppt I und LRA ausblenden)
- Sensoren: Loudness I (LUFS), Loudness LRA (LU)

Der Home-Assistant-Gerätename lautet OnAirScreen (Studio-1), wenn der Instanzname noch nicht im MQTT Device Name steht.

7.6 Bitfocus Companion (empfohlen)

Das eigene Companion-Modul **astrastudio-OnAirScreen** ist der empfohlene Weg für Stream Decks und ähnliche Oberflächen. Befehle gehen über HTTP (Port **8010**): GET /api/command?cmd=LED1:ON. Der Live-Status nutzt bevorzugt den OnAirScreen-**WebSocket** auf HTTP-Port **+1** (also **8011**, wenn HTTP 8010 ist) und fällt auf Polling von GET /api/status zurück (Standard alle 500 ms). **OSC ist nicht nötig**. Die AIR-Zeiten auf den Buttons bleiben nah am Studiobildschirm (MIC = AIR1). Beide Ports in der Firewall freigeben.

OnAirScreen: HTTP muss erreichbar sein (Standard). OSC kann aus bleiben.

Companion (lokales Modul, bis es im Companion Store liegt):

1. Im Repo OAS-Companion: yarn install && yarn build
2. Ordner als companion-module-astrastudio-onairscreen nach Companion module-local-dev verlinken (oder unter Companion **Developer** den Pfad eintragen)
3. Companion neu starten und Connection **astrastudio / OnAirScreen** anlegen
4. Host = OnAirScreen-IP, HTTP-Port 8010 (wie unter **Settings** → **Network**). WebSocket anlassen, außer du musst nur pollen.

Presets: LED1–4 (Toggle + Farbe), AIR1–4 mit Live-Beschriftung und Zeit auf dem Button (MIC = AIR1), TOH, Reset AIR3/4, NOW / NEXT / WARN, Silence, Loudness I+LRA, Reset I+LRA.

Variablen wie `$(oas:air1_time)`, `$(oas:lufs_i)` und `$(oas:lra)` und Feedbacks (LED an, AIR läuft, TOTH, Silence, WARN, Loudness I+LRA) kommen aus WebSocket oder Status-Poll. Connection-Label auf oas setzen, damit die Beispiele passen. Der Connection-Status zeigt Instanzname und Version, z. B. Studio-1 · 1.0.10. Die `HELP.md` des Moduls listet alle Aktionen, Feedbacks und Variablen.

Wenn kein eigenes Modul geladen werden kann, bleibt **Generic OSC** als Alternative (nächster Abschnitt). OSC-Status-Push ist langsamer (alle 5 Sekunden) als HTTP-Poll / WebSocket.

7.7 OSC (Port 8000)

OSC unter **Settings** → **Network** einschalten. Prefix ist `/oas`. Integer 1/0 bedeutet AN/AUS; ohne Argument = TOGGLE. Texte als String.

Setzen (Befehle):

Adresse	Argument	Funktion
<code>/oas/led{1-4}</code>	<code>i 0/1</code> , oder leer für Toggle	LED
<code>/oas/air{1-4}</code>	<code>i 0/1</code> , oder leer für Toggle	AIR-Timer
<code>/oas/air{3-4}/reset</code>	keines	AIR3/AIR4 zurücksetzen
<code>/oas/air3/toh</code>	<code>i 0/1</code>	Top-of-Hour
<code>/oas/air3/time</code>	<code>i Sekunden</code> (weglassen = Abfrage)	AIR3-Zeit setzen
<code>/oas/text/now</code>	<code>s Text</code>	NOW
<code>/oas/text/next</code>	<code>s Text</code>	NEXT
<code>/oas/text/warn</code>	<code>s Text</code>	WARN
<code>/oas/command</code>	<code>s COMMAND:VALUE</code>	Roh-API-Befehl
<code>/oas/lufs/integrated</code>	<code>i 0/1</code> , oder leer für Toggle	I + LRA starten/stoppen
<code>/oas/lufs/integrated/reset</code>	leer	I + LRA zurücksetzen

Abfrage (Antwort an den UDP-Absender, kein Send Host nötig): `/state`-Adresse senden (oder `/oas/status` für alles). Antworten sind Integer 0/1 für Booleans und Strings für Texte. `/oas/lufs/integrated/state` zeigt, ob I + LRA läuft. `/oas/lufs/i` und `/oas/lufs/lra` liefern I (LUFS) und LRA (LU) als String mit einer Nachkommastelle, oder leer wenn unbekannt.

Push (Companion-Feedback): OSC Send Host auf den Companion-Rechner, OSC Send Port auf den Generic-OSC-**Feedback**-Port. OnAirScreen sendet dann `/oas/led1/state` usw. nach Änderungen und alle 5 Sekunden.

```
# LED1 toggeln
python3 utils/oas_osc_send.py /oas/led1
```

```
# LED1 an
python3 utils/oas_osc_send.py /oas/led1 1
```

Alternative: Generic OSC

Wenn OSC bevorzugt wird oder das HTTP-Modul nicht geladen werden kann, eingebaute Connection **OSC Generic**.

Nur steuern:

1. OnAirScreen: OSC an, Listen-Port 8000
2. Companion: **OSC Generic**, Target = OnAirScreen-IP, Port 8000
3. Button-Action **Send integer**, Pfad /oas/led1, Wert 1 (an) oder 0 (aus). Ohne Argument = Toggle. **Send string** für /oas/text/now.

Button-Feedback (Stream-Deck-Farbe):

1. Companion: OSC-Generic-**Feedback-Listen-Port** setzen (z. B. 9000)
2. OnAirScreen: OSC Send Host = Companion-IP, OSC Send Port = dieser Feedback-Port
3. Feedback **Listen for OSC messages (Integer)** auf /oas/led1/state, Vergleich 1

Query-Reply an den Command-Socket nutzt Companion **nicht** (anderer UDP-Port).

7.8 SNMP (Port 1161)

SNMP unter **Settings** → **Network** einschalten. Der Agent spricht **SNMPv2c** und **SNMPv3** (USM, SHA-256, AES-128). Standard-Port ist **1161** (kein privilegierter Port). 161 ist möglich, braucht auf Linux/macOS aber Rechte.

Die MIB liegt in mibs/ASTRASTUDIO-ONAIRSCREEN-MIB.txt und zum Download im [Wiki](#). Enterprise-OID .1.3.6.1.4.1.66873 (IANA PEN 66873, astrastudio broadcast solutions). MIB-2 system (sysDescr, sysUpTime, sysName, ...) wird mitgeliefert.

GET / WALK (Status wie /api/status): LEDs, AIR-Timer, NOW/NEXT/WARN, Silence, Loudness I+LRA, Version, Instanz.

SET (gleiche Semantik wie OSC): LED-State 0/1/2 = OFF/ON/TOGGLE, AIR-State 0/1/2/3 = OFF/ON/TOGGLE/RESET, AIR3-Sekunden, AIR3-TOTH, Texte, oasLufsIntegrated, oder oasCommand mit einem Roh-COMMAND: VALUE. v2c-SET nur mit Write-Community. v3-SET, sobald ein Auth-Passwort gesetzt ist.

Traps: Silence, WARN-Textwechsel, optional LED-Wechsel. Version v2c oder v3 (v3 nutzt denselben USM-User).

```
# Status walk (v2c)
```

```
snmpwalk -v2c -c public localhost:1161 .1.3.6.1.4.1.66873
```

```
# LED1 an
```

```
snmpset -v2c -c private localhost:1161 .1.3.6.1.4.1.66873.1.2.1.2.1 i 1
```

```
# Walk mit SNMPv3 authPriv
```

```
snmpwalk -v3 -l authPriv -u nms -a SHA-256 -A 'auth-secret' -x AES -X  
'priv-secret' localhost:1161 .1.3.6.1.4.1.66873
```

Ohne Write-Community (Default) ist v2c nur lesend. noAuthNoPriv ist nur zum Testen gedacht.

7.9 Befehlsreferenz

Steuerungsbefehle

Befehl	Funktion
LED{1-4}: [ON/OFF/TOGGLE]	LED schalten
NOW: TEXT	NOW-Text setzen
NEXT: TEXT	NEXT-Text setzen
WARN: TEXT	Warnung setzen (Priorität 0)
WARN:1: TEXT	Warnung setzen (Priorität Medium)
WARN:2: TEXT	Warnung setzen (Priorität High)
WARN:	Warnung Priorität 0 löschen
WARN:1:	Warnung Priorität 1 löschen
WARN:2:	Warnung Priorität 2 löschen
AIR1: [ON/OFF/TOGGLE]	Mikrofon-Timer
AIR2: [ON/OFF/TOGGLE]	Telefon-Timer
AIR3: [ON/OFF/RESET/TOGGLE]	Radio-Timer
AIR3TIME: seconds	Radio-Timer auf Sekundenwert setzen
AIR3TOH: [ON/OFF/TOGGLE]	Top-of-Hour-Countdown
AIR4: [ON/OFF/RESET/TOGGLE]	Stream-Timer
LUFISI: [START/STOP/TOGGLE/RESET]	Programme-I + LRA-Session (Reset: laufend neu, gestoppt ausblenden)
FULLSCREEN: [ON/OFF/TOGGLE]	Hauptfenster Vollbild
CMD: REBOOT	OS-Neustart (HTTP: Web-Settings-PIN, wenn gesetzt)
CMD: SHUTDOWN	OS-Herunterfahren (HTTP: Web-Settings-PIN, wenn gesetzt)
CMD: QUIT	OnAirScreen beenden

Remote-Konfiguration (CONF)

Format: CONF:GRUPPE:PARAMETER=WERT

Änderungen werden erst nach CONF:CONF:APPLY=TRUE aktiv und gespeichert.

Befehl	Beschreibung
CONF:General:stationname=TEXT	Stationsname
CONF:General:instancename=TEXT	Instanzname (DNS-Label)
CONF:General:slogan=TEXT	Slogan
CONF:General:stationcolor=COLOR	Stationsfarbe
CONF:General:slogancolor=COLOR	Sloganfarbe
CONF:General:replacenow=[True/False]	IP-Ersatz aktivieren
CONF:General:replacenowtext=TEXT	Ersatztext
CONF:LED[1-4]:used=[True/False]	LED aktivieren
CONF:LED[1-4]:text=TEXT	LED-Text

Befehl	Beschreibung
CONF:LED[1-4]:activebgcolor=COLOR	LED-Hintergrund aktiv
CONF:LED[1-4]:activetextcolor=COLOR	LED-Text aktiv
CONF:LED[1-4]:autoflash=[True/False]	Autoflash
CONF:LED[1-4]:timedflash=[True/False]	20-Sekunden-Flash
CONF:Clock:face=FACE	Zifferblatt: digital, analog, analog_numbers, analog_studio, analog_railway, analog_24h_smooth, analog_24h_ticking (analog_24h ist Alias für Smooth)
CONF:Clock:digital=[True/False]	Digital / klassisch analog (Legacy)
CONF:Clock:showseconds=[True/False]	Sekunden anzeigen
CONF:Clock:secondsinoneline=[True/False]	Sekunden in einer Zeile
CONF:Clock:staticcolon=[True/False]	Statischer Doppelpunkt
CONF:Clock:digitalhourcolor=COLOR	Stundenfarbe
CONF:Clock:digitalsecondcolor=COLOR	Sekundenfarbe
CONF:Clock:digitaldigitcolor=COLOR	Ziffernfarbe
CONF:Clock:logopath=PFAD	Logo-Pfad
CONF:Clock:logoupper=[True/False]	Logo oben
CONF:Network:udpport=PORT	UDP-Port
CONF:Network:tcpport=PORT	HTTP-Port
CONF:Audio:enabled=[True/False]	Audio-Meter ein/aus
CONF:Audio:source=[device/livewire/aes67]	Audioquelle
CONF:Audio:input_device=DEVICE_NAME	Lokales Eingabegerät
CONF:Audio:livewire_channel=N	Livewire-Kanal
CONF:Audio:livewire_iface=IP_OR_EMPTY	AoIP-Interface-IP
CONF:Audio:aes67_id=ORIGIN_HASH	AES67-Stream-ID
CONF:Audio:aes67_addr=MULTICAST	AES67-Multicast
CONF:Audio:aes67_port=PORT	AES67-RTP-Port
CONF:Audio:aes67_name=NAME	AES67-Anzeigename
CONF:Audio:aes67_codec=[L16/L24]	AES67-Codec
CONF:Audio:aes67_rate=48000	AES67-Samplerate
CONF:Audio:aes67_channels=2	AES67-Kanalzahl
CONF:Audio:aes67_manual=[True/False]	AES67 Paste-SDP
CONF:Audio:unit=[dbfs/dbtp/bbc_ppm]	L/R-Anzeigeeinheit
CONF:Audio:layout=[lr/lufs/both]	Meter-Layout
CONF:Audio:display_style=[solid/bargraph]	Meter-Stil
CONF:Audio:meter_width=79	Meter-Breite (Pixel)
CONF:Audio:lufs_reference_preset=PRESET	LUFS-Preset
CONF:Audio:lufs_reference=-23.0	LUFS-Zielpegel
CONF:Audio:peak_hold=[True/False]	Peak-Hold ein/aus
CONF:Audio:peak_hold_seconds=1.5	Peak-Hold-Dauer
CONF:Audio:tooloud=[True/False]	TooLoud ein/aus
CONF:Audio:tooloudtext=TEXT	TooLoud-Text
CONF:Audio:tooloud_threshold_dbtp=-1.0	TooLoud-Schwellwert
CONF:Audio:tooloud_action=[warning/led]	TooLoud-Aktion
CONF:Audio:tooloud_led=[1/2/3/4]	TooLoud-LED

Befehl	Beschreibung
CONF:Audio:silence=[True/False]	Silence Detection
CONF:Audio:silence_warn=[True/False]	Silence-WARN ein/aus
CONF:Audio:silence_on_absent=[True/False]	Absent als Stille
CONF:Audio:silence_text=TEXT	Silence-WARN-Text
CONF:Audio:silence_threshold_dbfs=-50.0	Silence-Schwellwert
CONF:Audio:silence_duration_s=10.0	Silence-Dauer (s)
CONF:Audio:silence_recovery_s=2.0	Silence-Recovery (s)
CONF:Audio:silence_http_url=URL	Silence-HTTP-GET-URL
CONF:Timers:TimerAIR[1-4]Enabled=[True/False]	AIR aktivieren
CONF:Timers:TimerAIR[1-4]Text=TEXT	AIR-Label
CONF:Timers:TimerTOTHText=TEXT	TOTH-Timer-Label
CONF:Timers:AIR[1-4]activebgcolor=COLOR	AIR-Hintergrund aktiv
CONF:Timers:AIR[1-4]activetextcolor=COLOR	AIR-Text aktiv
CONF:Timers:AIR[1-4]iconpath=PFAD	AIR-Icon-Pfad
CONF:Timers:TimerAIRMinWidth=PIXEL	AIR-Mindestbreite
CONF:CONF:APPLY=TRUE	Konfiguration anwenden

CONF:Audio:unit=lufs wird weiterhin akzeptiert und setzt das Layout auf lufs (L/R-Einheit bleibt dbtp).

Farben: Hex-Format (#FF0000) oder Farbnamen.

8. Presets (Profile)

Presets ermöglichen das Speichern und Laden kompletter Konfigurationen.

Aktion	Schaltfläche	Beschreibung
Speichern	Save Preset...	Aktuelle Konfiguration als JSON-Datei speichern
Laden	Load Preset...	Gespeichertes Preset laden und anwenden
Löschen	Delete Preset...	Preset-Datei entfernen

Speicherort: <Konfigurationsverzeichnis>/presets/<name>.json

Presets enthalten Metadaten (Name, Version) und die vollständige Konfiguration als JSON.

Nach dem Laden muss **Apply** geklickt werden, damit die Einstellungen aktiv werden.

Hinweis: MQTT-Einstellungen werden in der Konfigurationsdatei gespeichert, sind aber **bewusst nicht** im Preset-Export enthalten. MQTT-Zugangsdaten sind installations- und umgebungsspezifisch und sollen nicht mit visuellen Profilen mitexportiert werden.

9. Kommandozeilenoptionen

```
python start.py --loglevel DEBUG
```

```
python start.py -l WARNING
```

```
python start.py --start-time 2026-09-14T12:00:00
```

Option	Werte	Beschreibung
<code>-l, --loglevel</code>	DEBUG, INFO, WARNING, ERROR, CRITICAL	Log-Level überschreiben (wird nicht gespeichert)
<code>--start-time</code>	ISO-8601 (<code>datetime.fromisoformat</code>)	Anzeige-Startzeit nur für diesen Lauf (wird nicht gespeichert)

Das Log-Level aus den Einstellungen (About/Log level) kann zusätzlich die Werte NONE (kein Logging) enthalten.

10. Konfigurationsspeicherort

Einstellungen werden über Qt QSettings gespeichert:

- **Organisation:** astrastudio
- **Anwendung:** OnAirScreen

Der genaue Pfad wird im Register **About** unter **Settings Path** angezeigt. Typische Speicherorte:

Plattform	Pfad
Linux	~/.config/astrastudio/OnAirScreen.conf
macOS	~/Library/Preferences/com.astrastudio.OnAirScreen.plist
Windows	Registry: HKEY_CURRENT_USER\Software\astrastudio\OnAirScreen

Logs und Crash-Reports liegen separat. Der genaue Pfad steht im Register **About** unter **Log Folder**:

Plattform	Log-Ordner
Linux	~/.local/share/astrastudio/OnAirScreen/logs/
macOS	~/Library/Logs/OnAirScreen/
Windows	%LOCALAPPDATA%\astrastudio\OnAirScreen\logs\

Dateien in diesem Ordner:

- `onairscreen.log` – rotierendes Anwendungs-Log (gleicher Inhalt wie `stderr`, folgt dem Log-Level)
- `crash-YYYYMMDD-HHMMSS.txt` – ungefangene Python-Exceptions (Traceback, Version, OS; keine Settings oder Passwörter). Nach einem Main-Thread- oder Qt-Fatal-Crash beendet sich die Anwendung; Fehler in Hintergrund-Threads werden nur geloggt.
- `fault.log` – native Abstürze (Segfaults in Qt oder C-Erweiterungen)

Crash-Dateien werden immer geschrieben, auch wenn das Log-Level `NONE` ist. Den Log-Ordner nur an den Support schicken, nicht öffentlich posten (DEBUG-Logs können Hostnamen oder Befehle enthalten). Nach einem Prozess-Absturz zeigt der nächste Start einen Dialog mit **Open log folder**; er schließt sich nach 30 Sekunden automatisch.

11. Fehlerbehebung

OnAirScreen startet nicht / Port belegt

- Prüfen, ob UDP-Port 3310 oder HTTP-Port 8010 bereits belegt ist
- Ports in **Network** ändern

Fernsteuerung funktioniert nicht

- Firewall-Regeln für UDP/HTTP/OSC/SNMP-Ports prüfen
- Korrekte IP-Adresse und Ports verwenden
- Mit curl `http://127.0.0.1:8010/api/status` lokal testen

GPIO schaltet LEDs oder Timer nicht

Raspberry Pi (Unter-Tab **Raspberry Pi**):

- Pi-GPIO funktioniert nur auf dem Raspberry Pi. Andere Plattformen zeigen auf diesem Unter-Tab „not available“
- Status unter **Settings** → **GPIO** → **Raspberry Pi** prüfen
- PC817-Modul verwenden; niemals 5–24 V vom Mixer direkt auf Pi-Pins
- Invert (Standard an für Kontakt nach GND) sowie Level- vs. Flanken-Modus prüfen
- GPIO und die jeweilige GPI-Zeile aktivieren, dann Apply

Axia Livewire (Unter-Tab **Axia Livewire**):

- Livewire GPIO und die Mapping-Zeile aktivieren, dann Apply
- Typische Mic-ON-Lampe: Richtung **GPO**, Pin **1**, Invert **aus**, Modus **level**
- **Multicast**: dieselbe Livewire-**Kanalnummer** wie das Audio der Source verwenden (GPIO advertised nicht). Rechner muss im AoIP-/Livewire-VLAN mit IGMP liegen; dasselbe Interface wie bei Audio Meters wählen. Status „join failed“: falsches Interface oder keine Multicast-Route
- **LWRP**: Mode auf **LWRP (TCP)**, Geräte-**Host** (TCP 93). Mapping-**Port** ist der GPIO-Port am Gerät, nicht ein Livewire-Audiokanal. Status „host is empty“ / „LWRP failed“: IP fehlt oder Gerät nicht erreichbar. VLAN gilt weiterhin; IGMP wird nicht genutzt
- Keine Pakete / LED bleibt aus: prüfen, ob Konsole/Gerät GPIO sendet, Kanal- oder Portnummer stimmt und OAS im Livewire-Netz ist
- Mic beim OAS-Start schon ON: OAS fordert nach Join/Connect den aktuellen Zustand an; bleibt die LED dunkel, prüfen, ob das Gerät darauf antwortet (oder den Kanal einmal toggeln)

Companion-Modul zeigt Disconnected

- HTTP muss auf dem konfigurierten Port erreichbar sein (Standard 8010); OSC nutzt dieses Modul nicht
- Mit curl `http://<OnAirScreen-IP>:8010/api/status` JSON prüfen

- Nach yarn build Companion neu starten, damit das lokale Modul neu geladen wird

Silence Detection löst nicht aus

- Ist **Enable Silence Detection** aktiviert und mit **Apply** gespeichert?
- Audioquelle prüfen (Device gewählt, Livewire-Kanal, AES67-Stream nicht **None**)
- dBFS-Schwellwert senken oder Duration verkürzen, wenn Restgeräusch über -50 dBFS liegt
- Fehlt Device oder Stream, **Trigger silence warning when Device/Stream is absent** einschalten

MQTT-Verbindung schlägt fehl

- MQTT-Broker erreichbar? (mqttserver, mqttport)
- Zugangsdaten korrekt?
- enable MQTT support aktiviert und mit **Apply** gespeichert?

NTP-Warnung erscheint dauerhaft

- Lokalen NTP-Server konfigurieren (NTP/ntpcheckserver)
- Bei Quelle Local die Systemzeit synchronisieren oder die Zeitquelle auf NTP/PTP umstellen
- NTP-Prüfung deaktivieren, falls nicht benötigt

Wetter-Widget zeigt nichts

- Gültigen OpenWeatherMap API-Key eingeben
- Städtenamen mit **Find** neben City ID suchen oder die ID manuell eingeben
- **Test API** in den Einstellungen ausführen

Livewire-Meter zeigt keine Pegel

- Source auf **Livewire** gesetzt und Channel-Nummer korrekt (oder announced Quelle gewählt)?
- Rechner im AoIP-/Livewire-VLAN? IGMP/Multicast nicht gefiltert?
- Passendes Netzwerk-Interface gewählt (nicht „Default“, falls mehrere NICs)?
- UDP-Port 5004 freigegeben?
- **Keine Namen unter Livewire Source:** Ads laufen auf 239.192.255.3:4001 nur bei offenen Settings (Desktop oder WebUI-Overlay). Die Kanalnummer kann trotzdem manuell eingegeben werden.

AES67-Meter: keine Streams oder keine Pegel

- **Keine Streams in der Liste:** AoIP-Interface, VLAN und IGMP prüfen; SAP ist 239.255.255.255:9875 und 224.2.127.254:9875 und läuft nur bei offenem Settings-Dialog oder WebUI-Overlay. Wenn das Gerät nicht announced, **Paste SDP** verwenden.
- **Stream da, Meter bleibt stumm:** RTP-Adresse/Port und Codec (L16 vs. L24). Dante muss im AES67-/SAP-Modus sein. PTP ist fürs Metering nicht nötig.

Einstellungen zurücksetzen

1. Einstellungsdialog öffnen (Ctrl+S)
2. Register **About** → **Enable Reset all settings button** aktivieren
3. **Reset all OnAirScreen settings to default** klicken
4. **Apply** klicken

Logs an den Support schicken

Wenn OnAirScreen abstürzt oder sich unerwartet verhält:

1. **Settings** → **About** öffnen
2. **Open log folder** klicken (oder den Pfad unter **Log Folder** kopieren)
3. `onairsteen.log` und vorhandene `crash-*.txt`-Dateien schicken (plus `fault.log`, falls nicht leer)

Die Reports enthalten keine Settings, MQTT-Passwörter oder API-Keys. `--loglevel DEBUG` nur kurz zum Reproduzieren nutzen, bevor die Logs verschickt werden. Nach einem Absturz erscheint dieser Dialog auch beim nächsten Start (schließt sich nach 30 Sekunden).

Anhang: Ereignisprotokollierung

OnAirScreen protokolliert intern folgende Ereignistypen:

- LED-Änderungen (Quelle: manual, autoflash, timedflash, API)
- AIR-Timer Start/Stopp/Reset
- Empfangene Befehle (UDP/HTTP/OSC/SNMP)
- Warnungen hinzugefügt/entfernt
- Einstellungsänderungen
- Systemereignisse (Start, Beenden, Neustart)

Das Log-Level steuert die Ausgabemenge. Bei Problemen empfiehlt sich temporär `--log level DEBUG`. Logs werden zusätzlich nach `onairscreen.log` im Log-Ordner geschrieben (siehe **About**).

© 2012–2026 Sascha Ludwig · astrastudio.de