

Companion — openRCS user guide

This module controls an Analog Way **LiveCore** (Ascender, NeXtage, SmartMatriX Ultra) or **Midra** (Pulse2, Eikos2, Saphyr, SmartMatriX2, QuickVu) video processor from a Stream Deck or any other Bitfocus Companion surface, over the switcher's native TCP control protocol on port **10500**.

It is the Companion companion to the [openrcs](#) control surface — the same reverse-engineered protocol, driven from a surface instead of a browser.

The [README](#) covers installing the module. This is how to build a page with it.

Before you rely on this: the wire protocol is covered by a dependency-free smoke test that checks the encoding byte for byte on both platforms, and the module has been loaded against the real Companion base library — a bank-aware take emits the correct direction and encoding, a master recall emits the right pair of lines, and the feedbacks resolve. **It has not been tested from Companion against real hardware.**

If the connection comes up with no actions, no variables and no presets at all, you are on a version that dies during startup — upgrade to the newest release before investigating anything else.

This module was built with AI assistance, directed and reviewed by a human author.

Configuring

- **Processor IP** — the switcher's address.
- **Platform** — LiveCore or Midra.

Pick the platform that matches the hardware, because it decides more than a label: LiveCore terminates commands with LF and Midra with CRLF, and the identity query differs. The wrong choice does not produce a helpful error; it produces a switcher that ignores you.

Two things about that port:

The control protocol is unauthenticated. Anyone who can reach port 10500 can drive the switcher. Keep it on a trusted network.

The device accepts very few concurrent control sessions. If openrcs-server or the vendor's own client already holds the connection, this module may simply not be able to connect. That is the switcher, not the module — close the other client.

Screens, groups and the 1-based count

"Screen/group" is **1-based**, and **an ungrouped screen is its own group**. So on a machine with no groups configured, screen 3 is group 3 and the numbering matches the front panel.

If you *have* configured groups, the numbering follows the groups, not the screens.

Take and Cut are bank-aware

This is the part worth understanding before you build a transport row.

The module **tracks each group's on-air bank and takes to the other one**. It does not send a fixed direction and hope. That is why the take direction is right after somebody else has driven the desk from the vendor's UI, and why the module needs its status query to be working before a take means anything.

Step back returns a group to its previous look. **T-bar** drives the transition bar manually across its full 0–65535 range, which is the action to attach to a fader.

The actions

Take / Cut — screen/group	Transition or cut, with the direction resolved from the tracked bank.
T-bar	Manual transition, 0–65535.
Step back	Previous look.
Recall master / screen memory	Recall and take, or load to preview only.
Freeze input	Per input.
Output black	Per output.
Raw command	A protocol line directly, mnemonic last — e.g. <code>1,50006Ctba</code> . Supports Companion variables.

Mnemonic last is the thing to remember about raw commands. The value comes first and is glued to the mnemonic with no separator. Getting it the other way round produces no error, just silence.

Feedbacks

- **On air (bank B / up)** — the group's up bank is live.
- **Transitioning** — a take is in progress.
- **Available** — the group holds an active screen.

Availability is worth putting on every transport button. A take against a group with no active screen is a press that does nothing, and it looks identical to one that worked.

Presets

Ready-made take and cut buttons for screens/groups 1–4 with on-air tally, and recall buttons for master memories 1–8. Build outward from those rather than from blank buttons — they already carry the tally wiring.

Troubleshooting

Symptom	Cause
No actions, variables or presets at all	The module died during startup. Upgrade to the newest release.
Cannot connect at all	Something else holds the switcher's control session — the vendor client, or openrcs-server. The device allows very few.
Commands are ignored, connection looks fine	Wrong Platform . The line terminator differs between LiveCore and Midra.
A take goes the wrong way	The tracked bank is stale. Check the connection has been up long enough to have read status.
A raw command does nothing	Mnemonic last, value first, glued: <code>1,50006Ctba</code> .
A transport button does nothing on one group	That group has no active screen. Add the <i>Available</i> feedback.

See also

- [README](#) — installing, and the full action/feedback/variable list
- [companion/HELP.md](#) — the same material, in Companion's help panel
- [openrcs](#) — the browser-based surface for the same protocol

Companion — openRCS · v1.0.1 · guide updated 2026-08-22 · stoatworks-labs.com/software/companion-openrcs/

Latest version of this guide: stoatworks-labs.com/software/companion-openrcs/guide/ · Source and issues:

<https://github.com/stoatworks-labs/companion-module-openrcs>