

KINEMATIC MIRROR MOUNT ■ USB & WI-FI CONTROL

# Command Reference Manual

Every command the mount accepts, over either transport — plus first-time setup, reconnecting over Wi-Fi, and the `physiverse` command-line client.

## 1 OVERVIEW

The mount has two independent axes, each a brushed DC motor with an analog Hall position sensor:

- **Axis 1** — motor M1, sensor a0
- **Axis 2** — motor M2, sensor a1

Commands are plain text, one per line, and are identical over both transports. In the syntax below, `N` is the axis number (1 or 2) and sensor values are raw ADC counts (0–4095).

| Transport  | Details                                                                                                                                                                                                                                           |
|------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| USB serial | 115200 baud, newline line endings. Always available; the only transport that accepts provisioning commands.                                                                                                                                       |
| WiFi (TCP) | Port 3333, discovered via mDNS as <code>&lt;name&gt;.local</code> , where <code>&lt;name&gt;</code> is the unit's name reported by status (unnamed units fall back to <code>mirrormount-&lt;id&gt;</code> ). Requires authentication (Section 6). |

Responses are text lines, mirrored to USB *and* the connected WiFi client.

## 2 MOUNTING ON A POST

The mount fixes to a standard post with an **8-32 set screw**. Seat it on the post, hold the mount by the **black plastic part** while you tighten, and stop as soon as it is held firmly.

▲ **WARNING** — Do not over-tighten. The body is a monolithic flexure structure, and excess clamping force can distort it and cause permanent damage. Tighten only by the black plastic part, and do not exceed **50 mN·m** of torque when fixing the mount to the post.

### 3 GETTING THE PHYSIVERSE CLIENT

Everything in this manual is driven by the `physiverse` command-line client, which lives in the companion repository `physiverse_monolithic_motorized_mirror_mount_client` (Python, macOS/Linux). **Start by cloning and installing it** — no `physiverse` command in this manual will run until you do:

```
git clone https://github.com/physiverse/\
physiverse_monolithic_motorized_mirror_mount_client.git
cd physiverse_monolithic_motorized_mirror_mount_client
pip install . # provides the "physiverse" command
python3 -m mount_client --help # or run it without installing
```

| Command                                                                                                                           | Description                                                                                                                                                                                                             |
|-----------------------------------------------------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <code>provision -ssid &lt;ssid&gt;</code><br><code>-wifi-password &lt;pw&gt;</code><br><code>[-device-password &lt;pw&gt;]</code> | First-time setup over USB (Section 4): saves WiFi credentials and, if given, the command password, then waits for the mount to join the network and prints its <code>.local</code> name.                                |
| <code>connect [&lt;host&gt; &lt;password&gt;</code><br><code>[tcp_port]]</code>                                                   | Interactive session over WiFi (authenticated). With <code>-usb</code> instead of <code>host/password</code> : serial session, no password needed. Ctrl-C or Ctrl-D quits; on USB every keystroke is forwarded as typed. |
| <code>status [&lt;host&gt; &lt;password&gt;]</code>                                                                               | One-shot report of the unit's name and both axes, plus WiFi state, over either transport (same transport selection as <code>connect</code> ).                                                                           |
| <code>factory-reset</code>                                                                                                        | Clear WiFi + command password over USB; axis calibration is kept.                                                                                                                                                       |

Transport selection, shared by `connect` and `status`: give `<host> <password>` for WiFi (TCP port defaults to 3333; `<host>` may be a `.local` name or an IP), or pass `-usb [PORT]` for serial — with no `PORT` the board is auto-detected (`/dev/cu.usbmodem*` on macOS); `-port PORT` is a synonym. `-baud` overrides the serial rate (default 115200).

```
physiverse status mirrormount-ac88.local mountpass # one-shot over WiFi
physiverse connect mirrormount-ac88.local mountpass # interactive over WiFi
physiverse connect --usb # interactive over serial
physiverse factory-reset # over USB only
```

#### Connecting over USB

USB is the transport that always works — before the mount has ever seen a network, and whenever you need it back. With the client installed, connect the board to your computer with a USB-C cable and run:

```
physiverse connect --usb # interactive session: type commands, Ctrl-C to quit
physiverse status --usb # one-shot report of name, axes and wifi, then exits
```

No password is needed over USB — authentication applies to WiFi clients only. The serial port is auto-detected (`/dev/cu.usbmodem*` on macOS); pass `-usb PORT` if more than one board is plugged in. Any serial console works just as well at 115200 baud (for example `pio device monitor` or `screen /dev/cu.usbmodem3101 115200`) — the commands are the same, you just type them yourself.

If the mount is new, continue with first-time setup below; if it is already provisioned, Section 5 covers reaching it over WiFi.

## 4 FIRST-TIME SETUP (OVER USB)

With the client installed (Section 3), the whole procedure is one command:

```
physiverse provision --ssid MyNetwork --wifi-password wifipass \
  --device-password mountpass
```

It auto-detects the serial port (`-port` overrides), stores the credentials on the device, waits until the mount joins the network, and prints the connect line for future use.

Manually, connect the mount by USB and open a serial console (e.g. `pio device monitor`). Then:

1. Set the device login password:  
`setpass <password>`
2. Give it your WiFi network:  
`wifi <ssid> <password>`
3. It confirms and prints its address, e.g.  
`wifi: CONNECTED ip=192.168.1.42 name=mirrormount-ab12.local`

Both are stored in flash and reconnect automatically on power-up — you should not need USB again for day-to-day use (Section 5). There are **two separate secrets**: the *login password* (`setpass`, authenticates clients) and the *WiFi credentials* (`wifi`, joins your network).

| Command                                         | Description                                                               |
|-------------------------------------------------|---------------------------------------------------------------------------|
| <code>setpass &lt;password&gt;</code>           | Set the login password (stored hashed). USB only.                         |
| <code>wifi &lt;ssid&gt; &lt;password&gt;</code> | Save WiFi credentials and connect. USB only. Password may contain spaces. |
| <code>wifi</code>                               | Show connectivity status (SSID, IP, name, signal).                        |
| <code>factory-reset</code>                      | Clear WiFi + password (keeps axis calibration). USB only.                 |

## 5 CONNECTING OVER WIFI LATER

Setup (Section 4) is done *once*. The credentials live in flash, so on every power-up the mount rejoins the same network by itself and starts listening on TCP port 3333. From then on a session is a single command — no USB, no re-provisioning:

```
physiverse connect mirrormount-ac88.local mountpass
physiverse status mirrormount-ac88.local mountpass # one-shot, then exits
```

The two arguments are the mount's address and the *login* password you chose with `setpass` (or `-device-password`) — not your WiFi password. Add a third argument only if the firmware was built to listen on a non-default TCP port: `physiverse connect mirrormount-ac88.local mountpass 4444`.

### Finding the address again

Prefer the `.local` name: it is the unit's own name, is printed at provisioning time, and never changes. The IP address comes from your router's DHCP lease and *can* change between power cycles, so treat it as a fallback rather than the thing you write down.

If you no longer know either one, ask the mount. Plug in USB and run:

```
physiverse status --usb
```

The report opens with the unit's name. A mount that answers mount: planksprank is reachable at planksprank.local. The trailing wifi line then gives the current SSID, IP, mDNS name and signal strength, e.g. wifi: CONNECTED ip=192.168.1.42 name=planksprank.local. The name is reported either way, so it is still there to read when the mount is *not* on the network; the SSID and IP appear only once it is connected. The same two lines come from typing status and wifi in a serial console, or in an already-open WiFi session. Failing that, look for the mount's name in your router's client list.

### When you do need USB again

Only for things that change the stored secrets or the network itself: re-run physiverse provision (or the wifi command) after moving the mount to a different network or changing that network's password; run setpass to change the login password; run factory-reset to clear both. Everything else — motion, calibration read-back, status — works over WiFi.

There is no password recovery: the login password is stored hashed, so a forgotten one is replaced over USB with setpass <new>, not read back.

### Troubleshooting

| Symptom                                                                        | Likely cause and fix                                                                                                                                                                                                                        |
|--------------------------------------------------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| cannot connect ...<br>Name or service not known                                | The .local name did not resolve. mDNS does not cross subnets and some corporate/guest networks block it — connect from the same LAN, or use the IP address reported by physiverse status -usb.                                              |
| cannot connect ...<br>Connection refused, or a timeout                         | The mount is powered but not on the network (or is on a different one). Check with physiverse status -usb; if it reports anything other than wifi: CONNECTED, re-provision.                                                                 |
| authentication failed<br>(wrong password?)<br>Connects, then drops mid-session | Wrong login password, or none was ever set. Set one over USB with setpass <password>.<br>Weak signal or the mount lost power. In-flight seekN/home moves still finish on their own; an open-ended manual drive is auto-stopped (Section 8). |

## 6 WIFI AUTHENTICATION

On each TCP connection the device issues a challenge; the client proves it knows the password without sending it:

```
device -> NONCE <hex>                (random, per connection)
client -> AUTH <hex>    hex = HMAC-SHA256(K, nonce)
device -> OK                (authenticated) -- or -- DENIED
```

where  $K = \text{SHA256}(\text{password})$ . After OK, the same commands as the serial console are accepted. The physiverse client (Section 3) performs this handshake for you.

## 7 CONTROL COMMANDS

Accepted on both transports (WiFi requires authentication first).

| Command                           | Description                                                                                                                                                                   |
|-----------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| seekN <target> <speed> [stall_ms] | Closed-loop move until sensor = target. speed is a magnitude; direction is automatic. Clamped to the calibrated range; gives up after stall_ms of no progress (default 2500). |
| homeN [speed]                     | Drive axis N to its home position (saved home, else range midpoint).                                                                                                          |
| home                              | Home <i>both</i> axes simultaneously.                                                                                                                                         |
| tickN < $\pm n$ > [spd] [ms]      | Fine adjustment: one continuous drive worth $n$ minimal ticks (sign of $n$ = direction). Pulse size defaults to the per-axis saved values (speed 100 / 5 ms unless changed).  |
| s                                 | Stop both motors.                                                                                                                                                             |
| r [n]                             | Read a0/a1, averaged over $n$ samples (default 1).                                                                                                                            |
| status                            | Report the unit's name (mount: <name>), then both axes: speed, polarity, range, home.                                                                                         |
| ?                                 | Help.                                                                                                                                                                         |

Calibration and low-level motor commands exist in the firmware but are intentionally not documented here — see the firmware repo's `PROTOCOL.md` if you know you need them (USB recommended).

## 8 NOTES

- **Calibration** (polarity, range, home) is per-axis, stored in flash, and survives reboot and factory-reset.
- seekN/home moves are self-terminating (range clamp + stall guard). If the WiFi client disconnects mid-move, they finish.
- Open-ended manual drives run until s; if the authenticated WiFi client is lost, this is the one motion that is auto-stopped (fail-safe).
- The sensor is non-linear near its travel extremes ("soft shoulders"); the calibrated range excludes them, and seek targets are clamped into it.

## 9 TYPICAL SESSION

```
# over USB, once:
setpass labpass123
wifi MyNetwork mypassword

# then, every session after that, over WiFi (or over USB):
physiverse connect mirrormount-ac88.local labpass123 ; or: physiverse connect --usb
status
home
seek1 2300 150
seek2 2100 150
tick1 -5          ; small nudge, axis 1, toward lower sensor values
r 16
```