

Forum:[Hack Forums](#)

Thread: [Physical Claude Code Status Light](#)

xadamxk - July 8, 2026, 8:47 pm

Blync for Claude Code

Turn an Embrava Blynclight into a Claude Code status light — red when it needs you, green when it's done.

[>> Download Latest Release <<](#)

What is it?

A small native app that turns an Embrava Blynclight into a real-world status indicator for Claude Code. Glance at the light instead of your terminal: it goes red the instant Claude needs your input, green when a turn finishes, and off while Claude is working.

It ships as one binary that plays three roles:

blync-cc — opens the settings window (pick device, colors, test, wire up hooks).

blync-cc daemon — the background daemon that holds the USB handle open and drives the light.

blync-cc notify <state> — records a state for the daemon. This is what Claude Code hooks call.

If a unit powers on but doesn't show up, it's usually a charge-only USB cable; swap in a data-capable one.

See it in action

? Requires attention — Claude is waiting on a permission prompt [\(Click to View\)](#)

? Finished recently — the turn just completed [\(Click to View\)](#)

Features

Live status at a glance — the light mirrors every stage of a Claude Code turn.

Instant red — driven by the `PermissionRequest` hook, so the light turns red the moment Claude is about to ask for approval (not after an idle delay).

Customizable appearance — per-state colors, adjustable brightness, and an optional "blink attention" mode.

Audio cues — optional tone with selectable style and volume.

One-click hook setup — merges the hooks into `~/.claude/settings.json` with a `.bak` backup written first. Fully idempotent and headless-friendly (`blync-cc configure-hooks`).

Upgrade-aware daemon — `daemon --ensure` detects a stale daemon from an older build, stops it (force-kills if wedged), preserves the current light state, and starts the fresh build. Rebuilds take effect on the next session automatically.

Resilient by design — any device error is non-fatal; the daemon keeps retrying so an unplugged or busy light never blocks Claude Code.

Tiny footprint — a single, stripped, LTO-optimized release binary. No installer, no runtime dependencies.

Status colors

Color	State	What it means
-------	-------	---------------

? Red	Requires Attention	Claude needs you — permission prompt / agent needs input.
-------	--------------------	---

? Yellow	Finished (Idle)	Claude is waiting on you — idle prompt / elicitation dialog.
----------	-----------------	--

? Green	Finished (Recently)	Claude finished responding — the turn is complete.
---------	---------------------	--

? Magenta	Error (Claude API)	
-----------	--------------------	--

An API / auth error ended the turn.

? Off

Working

Claude is actively working — you just sent a prompt, or a tool finished.

How the hooks map

Each moment of a Claude Code turn maps to a light state:

Situation

Claude Code hook

LED

Session startsSessionStartstarts / refreshes the daemon

You send a messageUserPromptSubmit? off (working)

Session endsSessionEnd? off

An approved tool finishedPostToolUse? off (working)

Claude needs you (instant)PermissionRequest? red

Claude needs you (backstop)Notification? red

Claude is waiting on youNotification? yellow

Turn completeStop? green

API / auth error ends the turnStopFailure? magenta

Notes

I doubt many people have a Blynclight but I thought I'd release this project incase anyone does or wants to pick one up.

I've been using this project personally and professionally for a few days now and love it.

github.com/xadamxk/claude-blync · MIT License

Pr0m3th3us - July 8, 2026, 9:10 pm

That is so cool. [Wake](#) any alternatives to the Embrava Blynclight that you know of that would work with this?

xadamxk - July 9, 2026, 2:40 pm

(Jul 8, 2026 09:10 PM)Pr0m3th3us Wrote: [CITE](#) That is so cool. [Wake](#) any alternatives to the Embrava Blynclight that you know of that would work with this?

There are a ton of lights like this on the market.

It wouldn't take much work with Claude to update the code for this to support others if you end up getting one.

Theoretically it should be easy to support an addition with an LED over serial too.

