



MIPI DSI Display + Backlight Bring-up

AN-004 · E1M and E1M-X SoMs

Document Number: AN-004 **Revision:** 0.3 **Date:** July 2026 **Status:** Preliminary

alplab.ai

© 2026 Alp Lab AB. All rights reserved.

1 Scope

Design an external MIPI DSI display and its backlight into a carrier board, and understand what the Alp SDK does and does not drive today.

Warning: No panel has been driven on real hardware through `<alp/display.h>`. The SDK's display class is code-complete and tested against an emulated display on `native_sim`; it has no silicon run. The E8 CDC200 + DesignWare MIPI-DSI bring-up path exists as a bench app but is **bench-unverified**, and its clock offsets were transcribed from the Alif device-family pack rather than measured. There is no V2N DSI backend at all. The hardware guidance in this note — lane counts, backlight sizing, pin assignment — is datasheet-backed and safe to design against. The software path is not yet a reproduced result.

Audience	Firmware engineers integrating an LCD / OLED display into the SoM design.
Prerequisites	QS- guide completed; MIPI DSI display panel with a 40-pin FFC or similar; panel-specific init sequence.
Outcome	A carrier-board display design you can fabricate, and a realistic picture of the firmware work in front of you.
Time	Design review: 30 minutes. Panel bring-up: budget bench time.
Source	<code>examples/display/lvgl-widgets-demo/</code> , <code>examples/display/lvgl-music-player/</code> in <code>alp-sdk</code> .

Table 1 Scope summary

2 Hardware Setup

SoM family	DSI interfaces + backlight
E1M-AEN	1 × 2-lane DSI (DSI0), up to FHD. DSI0 lanes 2 / 3 are present on the edge connector but unimplemented — leave them floating. On-module backlight driver (Kinetic KTD2801ECD-TR) supporting up to 10 LEDs in series or 2P6S, with a 36 V OVP threshold on its boost switch node. BL_LED_A (A31) anode, BL_LED_K (B31) cathode, BL_PWM shared with BL_LED_A.
E1M-X V2N / V2N-M1	1 × 4-lane DSI (DSI0), up to FHD. The RZ/V2N has one MIPI DSI instance: the DSI1 pins on the E1M-X edge connector are reserved and must be left floating. Same backlight driver block, but different pins — BL_LED_A / BL_PWM on B55, BL_LED_K on A55.

Table 2 DSI interfaces + backlight per family

Neither family provides on-module EMI or ESD protection on the DSI lanes; protect them with a low-capacitance TVS array on the carrier board.

The backlight LED current is set by a feedback resistor R_{FB} :

$$I_{LED} = 95 \frac{mV}{R_{FB}}$$

For a 20 mA LED chain: $R_{FB} = 95 \text{ mV} / 20 \text{ mA} = 4.75 \Omega$.

3 What the SDK Provides

`<alp/display.h>` is a **portable framebuffer surface**, not a DSI stack. It wraps Zephyr's `display_*` driver class — open, get-caps, blit, clear, blanking, close — and resolves whichever panel driver your devicetree binds via the `alp-display0..3` aliases. Understand what it is not:

- **There is no panel-init API.** The header exposes no DCS write path, no lane-count or timing configuration, and no DSI-specific surface of any kind. Panel init sequences live in the Zephyr panel driver for your controller, configured through devicetree – not passed in through the SDK.
- **There is no backlight API.** BL_PWM is driven as an ordinary PWM channel through `<alp/pwm.h>`. Nothing in the SDK ramps it for you, and there is no 50 % boot default.
- **There is no V2N DSI backend.** The portable Zephyr wrapper is the only real display backend that ships. Vendor backends for the V2N DSI / parallel-RGB path and the Alif LCD-IF path are both still pending.

`alp_gui_lvgl_attach()` (`<alp/gui.h>`) is a real LVGL v9 bridge: it creates an `lv_display_t` over any open `alp_display_t` and wires LVGL's flush callback to `alp_display_blit()`. It is code-complete and `native_sim`-tested against a test-double display backend; no real panel has been driven through it. RGB565 / RGB888 / ARGB8888 map to LVGL formats; `ALP_PIXFMT_MONO_VLSB` has no LVGL v9 equivalent and is refused.

Note: MIPI DBI Type C is a separate path, and not DSI. Zephyr's `zephyr,mipi-dbi-spi` controller with an SPI panel (ST7789V, ILI9341 and similar) resolves through the same `<alp/display.h>` surface, and is the path the SDK's own display examples exercise. Its coverage is a build-only `native_sim` scenario proving the devicetree / Kconfig / backend wiring – silicon bring-up is pending. If your product can accept an SPI panel, this is the shorter path today; do not read its status as evidence for DSI.

4 Software Walkthrough

```
west build -b <BOARD> examples/display/lvgl-widgets-demo
west flash
```

The example draws an LVGL widget grid onto whatever panel the board's devicetree binds, via `alp_gui_lvgl_attach()`. It does not configure DSI and does not touch the backlight.

For the E8 DSI chain specifically, `examples/aen/aen-dsi-display/` is the closest thing to a reference: it drives a Rocktech RK055HDMIPI4MA0 (720 × 1280, Himax HX8394 controller) through CDC200 → DesignWare MIPI-DSI → D-PHY → panel over 2 data lanes. Read it before starting your own bring-up, and read it critically:

- It is **bench-unverified**. Its devicetree lives in the example's own board overlay precisely because it has not earned promotion into the shared SoC devicetree, which happens only once bench-verified.
- The panel-native 62.346 MHz DPI pixel clock is **not exactly reachable** from the Alif 400 / 480 MHz clock tree – the nearest rates are roughly 66.67 / 60 MHz. The overlay carries the panel-native value; the achieved rate is a bench tunable you will have to resolve for your panel.
- Some D-PHY PLL reference clock IDs are still placeholders rather than real clock gates.
- The 720 × 1280 RGB565 framebuffer is 1.84 MB and must be pinned in SRAM0; it does not fit ITCM.

2D-GPU-accelerated blits use `<alp/gpu2d.h>`.

5 Troubleshooting

- **Blank screen, backlight on:** panel init sequence mismatch. The init sequence is the panel driver's, not the SDK's – check the DCS list in your Zephyr panel driver against the panel vendor's datasheet.
- **Core hangs during D-PHY setup:** a D-PHY configuration clock left ungated. The D-PHY config register block stalls the AXI bus if it is accessed unlocked.
- **Screen tearing:** TE (tearing-effect) line not connected or driver running without sync.
- **Backlight not lighting:** feedback resistor open / shorted; the on-module driver enters fault state. Check R_FB value and routing.
- **Image too dim:** BL_PWM duty too low. Nothing sets this for you; drive the channel from your application.

6 References

- **Examples:** `examples/display/lvgl-widgets-demo/`, `examples/display/lvgl-music-player/`, `examples/display/lvgl-benchmark/` in `alp-sdk`.
- **E8 DSI chain reference (bench-unverified):** `examples/aen/aen-dsi-display/` in `alp-sdk`.

- **SDK API:** `<alp/display.h>` (portable framebuffer surface), `<alp/gui.h>` (LVGL v9 binding), `<alp/gpu2d.h>` (2D GPU), `<alp/pwm.h>` (backlight PWM).
- **Silicon staging checks:** `examples/aen/aen-dsi-regcheck/` (devicetree bind only), `examples/aen/aen-gpu2d-bench/` in `alp-sdk`.
- **Hardware Design Guides:** **HG-AEN-001** §5.4, **HG-V2N-001** §6.5 (MIPI DSI + backlight design rules).
- **Datasheet DSI pin tables:** §2.3 in each datasheet.

7 Revision History

Revision	Changes	Date
0.1	Initial draft.	May 2026
0.2	Updated SDK references to current layout: example paths moved to <code>examples/display/</code> (<code>lvgl-widgets-demo</code> , <code>lvgl-music-player</code> , <code>lvgl-benchmark</code>); added <code><alp/display.h></code> / <code><alp/gui.h></code> / <code><alp/gpu2d.h></code> APIs and the <code>aen-dsi-regcheck</code> / <code>aen-gpu2d-bench</code> silicon reg-checks.	June 2026
0.3	Corrected two hardware errors: the RZ/V2N has one MIPI-DSI instance, not two (the guide previously implied a second DSI port that does not exist and that the V2N datasheet marks “leave floating”), and the backlight pins <code>BL_LED_A/BL_LED_K</code> given as A31/B31 are AEN-only – on V2N they are B55/A55, and A31 carries <code>PCIE1_RX0_P</code> . Retensed the display story to reality: no panel has been driven on real hardware through <code><alp/display.h></code> , which wraps Zephyr’s display class and exposes no panel-init, DCS, lane, timing or backlight API. E8 CDC200 + DesignWare DSI bring-up is bench-unverified; MIPI DBI Type C is a distinct, build-only path. <code>alp_gui_lvgl_attach()</code> is a real LVGL v9 bridge but is native_sim-tested with the real-panel bench pending. Removed a fabricated resolution claim. Fixed the Alp SDK repository links (<code>alpDevs</code> → <code>alplabai</code>).	July 2026

Table 3 Revision History