Raspberry Pi2 Screen issues

This summary captures the "perfect storm" of hardware and software conflicts you are facing. It is designed to be shared with developers or used as a reference for future troubleshooting.

1. Hardware Inventory

- **Computer:** Raspberry Pi 2 Model B (BCM2836 SoC).
- **Display:** KeDei 3.5" SPI LCD, **Version 3.0 (2015/12/01)**.
  - *Critical Fact:* This is a "Legacy" board that uses three **74HC595 shift registers** to mimic a parallel interface. It is NOT a standard SPI display.
- **Interface:** GPIO Pins (SPI Bus 0).

2. Software Environment

- **OS:** Raspberry Pi OS "Trixie" (Debian 13) 32-bit with Desktop.
- **Kernel:** 6.x (Modern Linux Kernel).
- **Python:** 3.13.
- **Drivers:** Transitioned from the old `sysfs` GPIO interface to the modern **Character Device (`gpiod`)** and `lgpio` architecture.

3. The Core Problem: "The Legacy Gap"

The fundamental issue is that **2015 hardware is trying to communicate with a 2025 operating system.**

- **GPIO Allocation Conflict:** In modern Debian/Trixie, the kernel "claims" pins via the Device Tree. When you try to use `RPi.GPIO` or `luma.lcd`, the OS throws an `lgpio.error: 'GPIO not allocated'` because it views the Python script as an intruder trying to hijack system-reserved SPI pins.
- **Timing Mismatch:** The 2015 shift registers require specific, slow timing (latencies) that were easily achieved on old kernels (like Jessie). Modern 64-bit/32-bit kernels use high-speed DMA (Direct Memory Access) for SPI, which sends data faster than the 74HC595 chips can process, resulting in a permanent **"White Screen."**
- **Driver Deprecation:** The original KeDei drivers were binary-blobs compiled for Kernel 3.x/4.x. These are physically incompatible with the Kernel 6.x architecture used in Trixie.

---

4. Technical Facts for Future Reference

Fact	Detail
Confirmed Working OS	Raspbian **Jessie** (Legacy Kernel 4.x).

Fact	Detail
Fail Symptom	Persistent **White Screen** (Backlight on, logic off).
Key Error Message	`lgpio.error: 'GPIO not allocated'` or `ImportError` for legacy bitbang classes.
Incompatibility	Modern `vc4-kms-v3d` graphics driver conflicts with SPI GPIO access.
Bus Speed Limit	Shift registers typically fail above **1MHz to 2MHz** on modern Pi hardware.
Pin Mapping (BCM)	`DC=9`, `RST=22`, `MOSI=10`, `SCK=11`, `CS=8`.

---

5. Recommended Paths Forward

If the "Zero Feed-in" automation is the priority, these are the only viable ways to view your data:

1. **The Web Dashboard:** Since you are on a Pi 2, running a lightweight Flask web server allows you to see the status on your phone or laptop without using any GPIO pins for display.
2. **HDMI Status Screen:** Since you have the "Desktop" version of Trixie, you can auto-launch a full-screen Python/Tkinter window on your HDMI monitor to show the power stats.
3. **Hardware Upgrade:** Replace the 2015 KeDei with a modern **I2C OLED (SSD1306)** or a **Waveshare SPI** display that has native driver support in the 2025 Linux kernel.

Would you like me to help you set up the "Auto-Launching HDMI Dashboard" since you already have the Raspberry Pi Desktop environment running?

Revision #1
Created 2025-12-23 17:16:57 UTC by Max
Updated 2025-12-23 17:17:19 UTC by Max