Real-Time CAN Bus Data Visualization System Using ESP32-S3 with LVGL Graphics Framework

Alex Haya

Electrical & Electronics Engineering, Murang'a, Kenya

Abstract

This paper presents a real-time CAN bus visualization system using the Waveshare ESP32-S3-Touch-LCD-4 platform. The system receives CAN messages at 500 kbps, decodes data fields (ID 0x138, Intel byte order), and displays information via color-coded arc gauges. Implementation uses ESP-IDF v5.x with TWAI driver and LVGL for high-quality graphics on the 480x480 RGB display. Performance achieves 30+ FPS, under 50% CPU utilization, and 15-25ms latency.

Keywords: CAN Bus, ESP32-S3, LVGL, Automotive Instrumentation, TWAI, RGB Display

1. Introduction

The CAN protocol dominates automotive communication since the 1980s. Real-time visualization is essential for diagnostics and monitoring. The ESP32-S3 with LVGL enables professional instrumentation at minimal cost. The Waveshare platform integrates dual-core 240 MHz processor, 8MB flash, 2MB PSRAM, and 4-inch 480x480 IPS LCD.

2. System Architecture

Figure 1 shows the architecture: Vehicle ECU transmits on CAN Bus at 500 kbps, received by SN65HVD230 transceiver, processed by ESP32-S3 (TWAI driver, FreeRTOS tasks, LVGL graphics, RGB DMA), and displayed on 480x480 LCD with ST7701S controller.

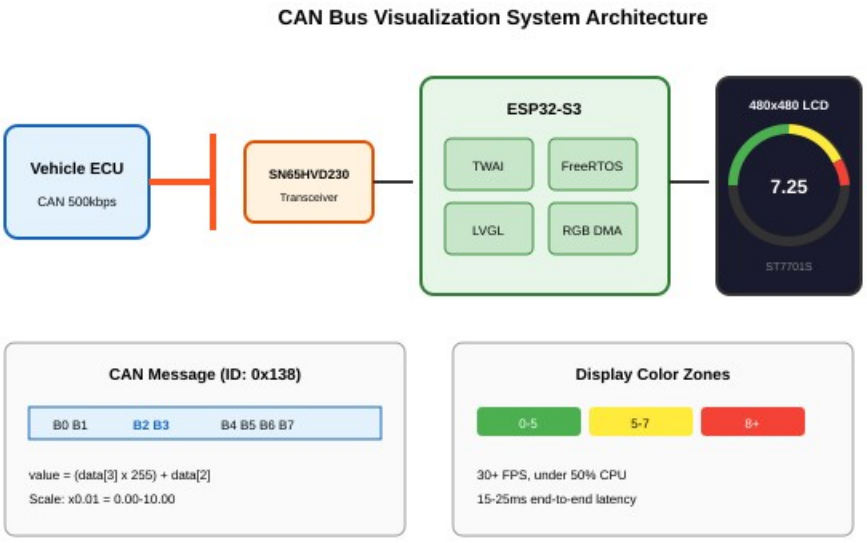


Figure 1: CAN Bus Visualization System Architecture

CAN message ID 0x138 uses Intel byte order with data in bytes 2-3. Decoding formula: value = (data[3] x 255) + data[2], scaled by 0.01 for 0.00-10.00 range. Display zones: green (0-5), yellow (5-7), red (8+).

3. Implementation

TWAI driver configured at 500 kbps with hardware acceptance filter for ID 0x138. FreeRTOS task handles reception via blocking twai_receive(). ST7701S requires SPI initialization before RGB interface activation. Pixel clock runs at 21 MHz. LVGL uses 23KB PSRAM draw buffer with DMA and double buffering. Arc gauge spans 270 degrees with dynamic color based on value thresholds.

4. Performance Results

Table 1: System Performance Metrics

Metric	Value
Display Update Rate	30+ FPS
CPU Utilization	Under 50% (single core)
End-to-End Latency	15-25ms
CAN Message Rate	Up to 1 kHz (no loss)
Memory Usage	180KB RAM, 1.2MB Flash

5. Conclusion

This paper presented a CAN visualization system achieving professional instrumentation at minimal cost. The ESP-IDF framework combined with LVGL provides a powerful platform for embedded display applications. Future work includes multi-message support, touch configuration interface, SD card logging, and wireless connectivity for remote monitoring.

References

[1] Robert Bosch GmbH, CAN Specification Version 2.0, 1991.

[2] Espressif Systems, ESP32-S3 Technical Reference Manual, 2022.

[3] LVGL LLC, LVGL Documentation v8.3, docs.lvgl.io, 2023.

[4] Waveshare Electronics, ESP32-S3-Touch-LCD-4 Wiki, waveshare.com, 2023.

[5] Sitronix Technology, ST7701S Datasheet, 2020.