Bottle Sumo Real-Time Streaming Firmware Architecture and Sensor Timing Report

CTEA Bottle Sumo Project

October 5, 2025

Abstract

This report documents the dual-core firmware that powers the Bottle Sumo real-time streaming robot. It consolidates the execution responsibilities of Core 0 and Core 1, explains the shared-data synchronization strategy, and quantifies the timing budget for each sensor pipeline. The firmware reference for all observations is `BottleSumo_RealTime_Streaming.ino` (rev. October 5, 2025).

1 Firmware Overview

The RP2040 application splits responsibilities across its two cores:

Core 0 — Control and Streaming Initializes the I^{2}C bus, drives Wi-Fi access point and TCP telemetry (§4), updates the OLED dashboard, and evaluates tactical decisions every 10 ms (`loop()`, lines `933--1012`).

Core 1 — Sensor Acquisition Services the ADS1115-based QRE reflectance array and the VL53L0X distance sensors, publishing fresh samples into `SharedSensorData` via mutex-guarded writes (`loop1()`, lines `1017--1041`).

The runtime shares measurement results and configuration through a trio of mutexes: `data_mutex` protects sampled values, `wire1_mutex` serializes OLED and ToF transactions on `Wire1`, and `threshold_mutex` guards the adjustable edge-detection voltage (lines `118--153`).

2 Architecture Diagram

Figure 1 summarises the major execution blocks, buses, and data transfer paths. The diagram originates from the Mermaid source in `docs/bottle_sumo_architecture.mmd` and can be regenerated with the Mermaid CLI.

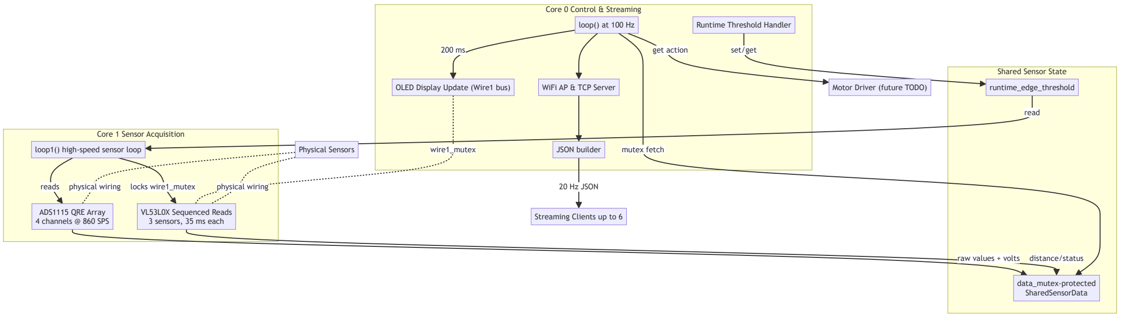

Figure 1: High-level dual-core architecture and data flows.

3 Sensor Pipelines and Timing

The firmware budgets sensor time explicitly to guarantee that control and telemetry loops meet their deadlines. Table 1 aggregates configuration constants and measured latencies.

Table 1: Sensor timing budget

Pipeline	Configuration Reference	Cycle Time	Notes
IR Reflectance (ADS1115)	`RATE_ADS1115_860SPS` (line `520`)	1.16 ms per channel 4.6 ms for full burst	Four channels sampled sequentially inside `updateSharedIRData()` (lines `1057--1078`); ADS conversion sets the throughput ceiling.
ToF Distance (VL53L0X x3)	`TOF_TIMING_BUDGET_US = 35 000` (lines `37--44`)	35 ms per sensor 115 ms per sweep	Sensors are read sequentially under `wire1_mutex` in `readToFSensors()` (lines `601--676`); extra 10 ms slack accounts for mutex and copy overhead.
Shared Data Publication	`TOF_DATA_FRESHNESS_MS = 150` (line `50`)	115 ms ToF freshness window 6 ms IR freshness window	`loop1()` pushes IR on every iteration and ToF when `now - lastToFUpdate` > `115 ms` (lines `1020--1041`).
Control Loop (Core 0)	`delay(10)` (line `1009`)	10 ms cadence	Each iteration validates data freshness, updates streaming clients, refreshes OLED every 50 iterations, and evaluates tactical state.
Telemetry Stream	`STREAM_SEND_INTERVAL = 50 ms` (lines `159--165`)	50 ms transmit cadence	JSON packets include core frequencies, IR thresholds, and ToF direction hints (lines `861--912`).

The ToF cycle demonstrates the longest blocking section. Combining the constants yields Equation (1):

$$t_{\text{ToF cycle}} = 3 \times 35\,\text{ms} + 10\,\text{ms} \approx 115\,\text{ms}, \tag{1}$$

which aligns with the freshness tolerance enforced by `TOF_DATA_FRESHNESS_MS`.

4 Streaming and Telemetry

Core 0 throttles TCP maintenance to 40 Hz and emits telemetry at 20 Hz. Each payload bundles IR voltages, ToF metrics, and inferred tactical state (`sendRealTimeStreamToAllClients()`, lines `861--924`). The OLED dashboard is synchronized with the same shared data while respecting the I^{2}C mutex (`updateOLEDDisplay()`, lines `680--759`).

5 Verification Checklist

- Sensor acquisition split across cores; shared mutexes prevent I^{2}C contention.
- IR loop provides >150 Hz updates, matching the design banner displayed at boot (line `707`).
- ToF loop maintains ≤115 ms age, satisfying `TOF_DATA_FRESHNESS_MS` (line `50`).
- Streaming cadence fixed at 20 Hz, with Wi-Fi housekeeping decoupled to avoid blocking control logic.

6 Reproduction Notes

1. Regenerate the Mermaid SVG/PDF: `mmdc -i docs/bottle_sumo_architecture.mmd -o docs/bottle_sumo_architecture.pdf`.
2. Compile this report: `tectonic docs/bottle_sumo_report.tex` (or any LaTeX engine with `graphicx` and `siunitx`).

7 Appendix: Key Firmware Constants

- Wi-Fi AP configuration: lines `72--105`.
- Streaming intervals and throttles: lines `138--168`.
- Sensor data freshness thresholds: lines `169--184`.
- Edge detection thresholds: lines `190--207`.