**Sensor Data Sharing and CLI Command Access Design Document**

**Overview**

This document outlines the design of a FreeRTOS-based system using CMSIS-RTOS2 API, where sensor data is shared between tasks using a global structure protected by a mutex. The data is updated periodically by SensorTask and accessed on-demand via CLITask when the user requests sensor data through the UART-based command-line interface (CLI).

**Goals**

* Provide thread-safe access to shared sensor data.
* Allow CLI to read the latest sensor values on-demand.
* Maintain separation of concerns: SensorTask handles acquisition, CLITask handles user interaction, LoggerTask handles UART output.

**Tasks and Responsibilities**

SensorTask periodically reads the sensor values and updates a shared global structure protected by a mutex.  
LoggerTask handles UART logging using a queue mechanism. It prints data passed by other tasks.  
CLITask waits for user commands via UART and executes matching commands like 'get\_sensor', 'version', 'reboot'. It accesses sensor data through a mutex-protected shared structure.  
OTATask is reserved for actual firmware update handling and is not used for CLI.  
HeartbeatTask toggles a GPIO or logs a heartbeat message periodically.

**Shared Sensor Data Access**

A global structure stores temperature, humidity, and a timestamp.  
This structure is protected using an osMutex (CMSIS-RTOS2) to avoid race conditions.  
SensorTask updates this structure every 1 second.  
CLITask reads it when the user runs 'get\_sensor' command.

**CLI Command Handling**

handle\_command() is triggered when the user presses Enter (\r or \n). It processes the buffered command string.  
For 'get\_sensor', CLITask acquires the mutex, reads the latest sensor snapshot, and sends it to LoggerTask for UART printing.

**Design Benefits**

* Ensures safe, concurrent access to sensor data.
* Decouples responsibilities between acquisition, logging, and user interaction.
* Allows scalable CLI commands.

**Future Enhancements**

* Wrap mutex access into a helper function like sensor\_get\_snapshot().
* Add CLI commands like 'set\_interval', 'get\_status', etc.
* Implement timestamp formatting for logs.

Document Version: 1.0  
Author: Halak Vyas  
Date: July 7, 2025