Design Report: Real-Time Railway Crossing Controller

---

1. Project Overview

The Real-Time Railway Crossing Controller is designed using an STM32F446VET6 microcontroller, aimed at managing railway crossing barriers using real-time sensor inputs, control logic, and motor/relay actuation. The design leverages FreeRTOS to manage tasks and ensure responsive, modular operation.

---

2. System Components

- **Microcontroller**: STM32F446VET6
- **RTOS**: FreeRTOS
- **Peripherals**:
  - RTC (Real-Time Clock)
  - Sensors (IR/Proximity sensors)
  - Relay/Motor Driver (Barrier control)
  - UART/Storage (Data logging)

---

3. System Architecture

The system follows a modular multi-tasking architecture with clearly separated roles:

- **Sensor Task**
  - Polls or is triggered by the RTC
  - Reads sensor data (e.g., train arrival/departure detection)
  - Sends messages to the Control Task via a Message Queue
  - Reports to the Watchdog
  - Sends log messages to the Log Task
- **Control Task**
  - Receives messages from the Sensor Task
  - Executes control logic (e.g., open/close barrier)
  - Interfaces with the Relay/Motor Driver
  - Optionally logs control actions
- **Log Task**
  - Receives logs from Sensor and Control tasks
  - Stores data via UART or onboard storage
- **Watchdog**
  - Monitors Sensor Task
  - Can reset or flag failures in task execution

---

4. Communication and Synchronization

- **Message Queue**:
  - Used for Sensor-to-Control communication
  - Decouples sensing from decision-making
- **Logging Pipeline**:
  - Asynchronous log queue from Sensor and Control tasks to Log Task
- **Watchdog Monitoring**:
  - Periodic ping mechanism from Sensor Task

---

5. Timing Configuration

- **Timer (TIM2)** configured as follows:
  - Prescaler: 8399
  - Period: 500
  - Generates 50ms tick (for scheduling/logging)
  - Interrupt priority: 5 (suitable for application-level timing)

---

6. Real-Time Behavior

- Sensor readings and RTC triggering are deterministic
- Message Queue avoids blocking Control Task
- Control operations (e.g., barrier actuation) are non-blocking and interrupt-safe
- Logging is offloaded to a separate task to avoid jitter

---

7. Error Handling and Robustness

- Watchdog detects and mitigates task failures
- Timeout mechanisms in queue and sensor polling
- Logs maintain traceability of state and failures

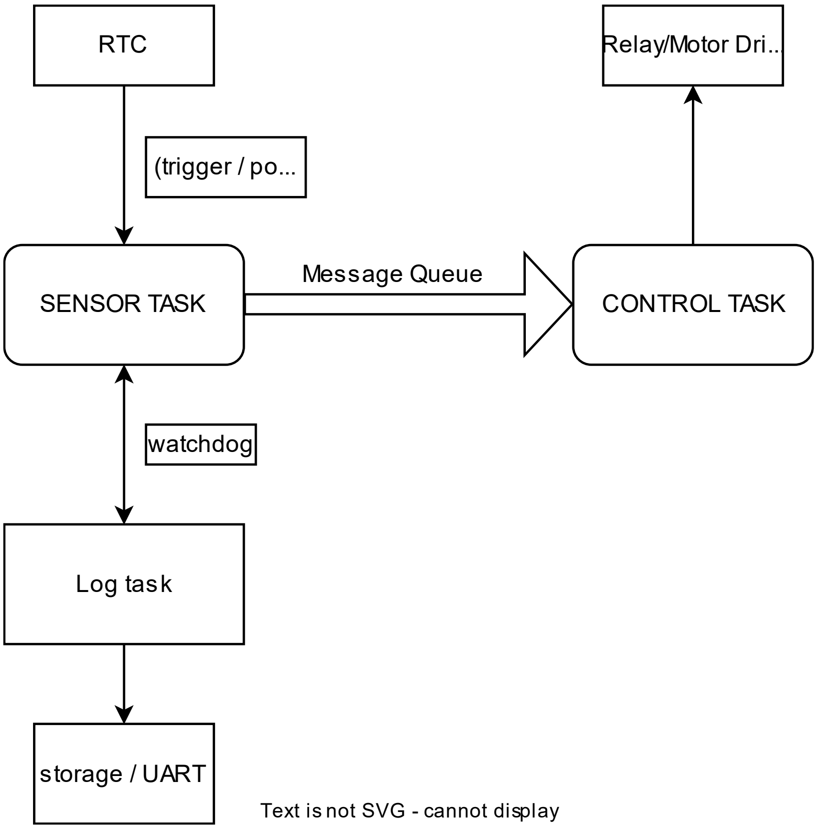
---

8. Future Improvements

- Introduce Over-The-Air (OTA) logging or wireless data offload
- Integrate external watchdog IC for hardware-level fault recovery
- Add diagnostic LED status indicators

---

8. Diagram


RTC
Relay/Motor Dri...
(trigger / po...
SENSOR TASK
Message Queue
CONTROL TASK
watchdog
Log task
storage / UART
Text is not SVG - cannot display