Software Requirement Specification (SRS)

Project Name: RTOS-Based Traffic Accident Detection System

Table of Contents

RTOS-Based Traffic Accident Detection System

1.0 Title of the Project

RTOS-Based Traffic Accident Detection System

2.0 Introduction

This document outlines the design and implementation of an RTOS-based Traffic Accident Detection System using an ESP32 microcontroller. The system employs vehicle sensors to detect accidents and automatically alerts emergency services with real-time GPS location and accident severity data. This system aims to improve response times and enhance road safety.

2.1 Purpose

The purpose of this project is to design and develop an accident detection system that operates in real-time, using an RTOS on an ESP32 microcontroller. It aims to provide quick and accurate accident detection and reporting, ensuring emergency services are informed immediately with essential details such as location and severity of the incident.

2.2 Scope

This system is intended for installation in vehicles to automatically detect and report traffic accidents. It utilizes multiple sensors to monitor conditions and identify accident events. The system can be deployed in personal, commercial, and public transportation vehicles to enhance road safety. It supports real-time GPS tracking and wireless communication with emergency services.

2.3 References

- **Reference 1**: ESP32 wroom microcontroller datasheet and technical reference manual.
- **Reference 2**: RTOS documentation and development guides (e.g., FreeRTOS, platform).
- **Reference 3**: Industry standards on accident detection and emergency response protocols.

3.0 Overall Description

3.1 System Perspective

The RTOS-based Traffic Accident Detection System is designed to operate as an embedded system within a vehicle. It integrates an ESP32 WROOM microcontroller with various sensors to detect accidents. When an accident is detected, the system captures GPS location and severity data and transmits this information to emergency services. The RTOS allows for efficient task scheduling and multitasking to ensure real-time processing.

3.2 System Functions

- **Accident Detection**: Detects accidents using input from accelerometers, gyroscopes, and impact sensors.
- **Emergency Alert**: Sends real-time alerts to emergency services with GPS location and severity data.
- **Data Logging**: Logs incident data for post-accident analysis.
- **System Self-Check**: Periodically checks the health of sensors and communication modules.
- **RTOS Task Management**: Manages tasks for data acquisition, processing, and communication in real-time.

3.3 Operating Environment

The system is intended to operate in vehicle environments that may experience extreme temperatures, vibrations, and fluctuating power conditions. The Esp32 microcontroller and sensors are selected to withstand these harsh conditions and maintain reliable performance.

3.3.1 Server

- **Operating System**: Real-Time Operating System (RTOS) such as FreeRTOS or CMSIS-RTOS.
- **Programming Language**: C/C++ for firmware development.
- **Microcontroller**: ESP32 series

3.4 Design and Implementation Constraints

- **Real-Time Processing**: The system must detect and report accidents within seconds to ensure timely emergency response.
- **Hardware Constraints**: Limited processing power, memory, and input/output pins of the STM32 microcontroller.
- **Environmental Factors**: The system must function reliably in variable weather, temperature, and vibration conditions.
- **Power Efficiency**: The system must consume minimal power to prevent draining the vehicle’s battery.

3.5 User Documentation

User documentation will include setup guides, system architecture details, RTOS task descriptions, and support guides. These resources will enable operators to install, configure, and maintain the system.

4.0 External Interface Requirements

4.1 User Interfaces

- **Emergency Alert System**: Interface to connect with emergency services through SMS, email, or a dedicated server.
- **User Configuration Interface**: Mobile app or web dashboard for configuration and monitoring.

5.0 System Features

- Real-Time Accident Detection
- Automatic GPS Location Tracking
- Accident Severity Analysis
- Emergency Alert Transmission
- Real-Time Data Logging and Reporting
- Low-Power Operation and Self-Diagnostic Checks

6.0 Deliverables

- **Hardware Components**: Esp32 microcontroller, accelerometers, gyroscopes, GPS module, and GSM/GPRS or LTE module for communication.
- **Software Components**: RTOS-based firmware for data collection, processing, and communication.
- **Documentation**: User manuals, technical documentation, and RTOS development guides.
- **Test Reports**: Functionality tests, performance evaluations, and user acceptance tests.

7.0 Hardware Design

- **Microcontroller**: ESP32 WROOM to handle sensor data, process events, and manage communication.
- **Sensors**: Accelerometer, gyroscope, impact sensor for accident detection.
- **GPS Module**: Provides location data to be sent with emergency alerts.
- **GSM/GPRS or LTE Module**: Sends emergency messages to authorities.
- **Power Supply**: Vehicle battery connection with voltage regulators and protection circuits.

8.0 Software Design

RTOS Tasks:

- **Sensor Data Acquisition Task**: Collects sensor data and filters noise.
- **Accident Detection Task**: Uses an algorithm to determine accident occurrence and severity.
- **GPS Data Acquisition Task**: Collects GPS coordinates.
- **Emergency Alert Task**: Sends alert messages to emergency services.
- **System Health Check Task**: Monitors hardware and software health.

9.0 References and Datasheets

- **STM32 Datasheet**: Link to official Esp32 datasheet
- **RTOS Documentation**: FreeRTOS Documentation
- **Sensor Datasheets**: Manufacturer-provided datasheets for accelerometers, gyroscopes, and GPS modules.