6. Software Design Description

6.1. Introduction

This document presents the architecture and detailed design for the software of the FSAE Vehicle Control System project. The project aims to develop a comprehensive control system for a Formula SAE vehicle, encompassing data acquisition, processing, and real-time telemetry.

6.1.1. System Objectives

The primary goal is to create a robust and efficient control system that enhances vehicle performance and safety. This includes developing modules for data management, input observation, telemetry, and various control units such as traction and suspension.

6.1.2. Hardware, Software, and Human Interfaces

- Hardware Interfaces:
    - The project uses Raspberry Pi units to handle processing for various control systems, while Arduinos are used to gather data from sensors for inputs like steering and pedal positions. The system also includes brushless DC motors for suspension control, which help optimize the vehicle's handling. LCD displays are used in the cockpit to provide telemetry data to the driver.
- Software Interfaces:
    - The software runs on both a Windows laptop for the database and telemetry GUI and on the Raspberry Pi units in the car. The database, managed in PostgreSQL, stores telemetry data. The Telemetry GUI, developed in Go with the Fyne.io framework, allows users to view real-time data. Additionally, the system’s core control functions are programmed in C++ to manage tasks like traction control and suspension adjustments.

- Human Interfaces:
    - The driver and team interact with the system through two primary interfaces. The telemetry GUI provides the race team with real-time performance and diagnostics data, allowing for strategic adjustments and troubleshooting. Meanwhile, the driver uses the steering wheel interface to receive immediate feedback on key metrics like speed, tire temperatures, and lap times, which is crucial for making quick decisions on the track.

6.2. Architecture Design

6.2.1. Major Software Components

6.2.1.1. **Database CSC:**

6.2.1.1.1. Manages and stores telemetry data collected from the vehicle.

6.2.1.2. **Input Observer CSC:**

6.2.1.2.1. Captures and processes input data from the vehicle's steering and pedals.

6.2.1.3. **Data Management CSC:**

6.2.1.3.1. Controls data flow between various systems within the vehicle.

6.2.1.4. **Telemetry GUI CSC:**

6.2.1.4.1. Provides a graphical interface for real-time data visualization.

6.2.1.5. **Traction Control CSU:**

6.2.1.5.1. Manages wheel traction to prevent slippage.

6.2.1.6. **Virtual Differential CSU:**

6.2.1.6.1. Controls power distribution between the vehicle's drive wheels.

6.2.1.7. **Suspension Motor Controller CSC:**

6.2.1.7.1. Manages the vehicle's active suspension system.

6.2.1.8. **Steering Wheel Unit Processor CSC:**

6.2.1.8.1. Processes driver inputs and displays data on the steering wheel interface.

6.2.1.9. **Safety Check CSC:**

6.2.1.9.1. Monitors system health and triggers safety responses.

6.2.1.10. **Power Draw Monitor CSC:**

6.2.1.10.1. Tracks and monitors power consumption of various systems.

6.2.1.11. **Graphics Interface CSC:**

6.2.1.11.1. Displays crucial telemetry and system data to the driver.

6.2.1.12. **Suspension Kinematics CSU:**

6.2.1.12.1. Calculates optimal suspension dynamics based on driving conditions.

6.2.2. Major Software Interactions

- Sensor Data Collection:
  - Multiple sensors send analog data to the CCU through an I2C bus. The data collected includes inputs such as GPS, tire temperatures, and acceleration, all essential for vehicle control.
- Active Suspension, Steering, and Motor Control Communication:
  - The Active Suspension Controls, Steering/Pedal Unit, and Motor Control Units are directly connected to the CCU via cabled connections. This setup enables continuous, direct exchange of commands and data, which allows these control units to receive updates and send feedback to the CCU.
- Wireless Steering Wheel Interface:
  - The Steering Wheel Unit communicates with the CCU through a WiFi connection. This wireless link allows the driver's inputs from the steering wheel to be sent to the CCU, while telemetry and feedback data are transmitted back to the steering wheel display for real-time monitoring.
- Data Logging and Transmission to the Database:
  - The CCU sends processed telemetry data to an external database over WiFi. This data includes real-time metrics like tire pressure, engine metrics, and speed. The stored data can then be accessed by the Telemetry GUI for live viewing and further analysis by the race team.

6.2.3. Architectural Design Diagrams

FSAE_Software_UML_Detailed_Light.png

- https://drive.google.com/file/d/1XJQdLfvUPEc-hL9Wth4m16wxI5Ti94fA/view?usp=sharing

FSAE_Software_UML_TopView.png

- https://drive.google.com/file/d/1ZSpapb2ohz1bcl9hZ1t0WoXGupmw0adg/view?usp=sharing

FSAE_Software_UML_UseCase.png

- https://drive.google.com/file/d/1l0SlbYij5DJVT3_ixCevE5P6Q_ZJQMkD/view?usp=sharing