
Pratik Patel - ppate460 & Uvaish Bakaliya - ubaka2

Group Number: 39

Smart Traffic Control System

- Our project focuses on developing a Smart Traffic Control System using Arduino microcontrollers, aiming to optimize urban traffic management. By integrating radar sensors with traffic signals, the system dynamically adjusts signal timing based on vehicle presence, reducing congestion and enhancing efficiency.
- We employ multiple Arduinos for data collection and central control, utilizing Serial Communication for inter-Arduino communication.
- Inputs include push buttons for pedestrian crossings, while outputs utilize LEDs for signal indication. Original work lies in the innovative integration of Arduino components and wired communication protocols.

Project Idea

- The Smart Traffic Control System revolutionizes urban traffic management by dynamically adjusting signal lights based on vehicle presence detected by radar sensors.
- This system aims to reduce traffic congestion and improve efficiency in urban transportation.

Input/Output Devices Used:

- Inputs for pedestrian crossings usually include **push buttons** that pedestrians use to request safe passage across the street.
- Conversely, outputs from this system are typically manifested through **Light-Emitting Diodes (LEDs)** integrated into the traffic signals.
- Additionally, we use **radar sensors** to detect incoming traffic and adjust the behavior of the LEDs (i.e., signals) based on the traffic conditions.

Communication Used between Arduinos:

- We are establishing communication between Arduinos using Serial Communication (UART). This method offers simplicity, reliability, and efficient data transmission, making it an ideal choice for inter-Arduino communication in our project.
- By leveraging UART, we ensure seamless coordination and exchange of information between multiple Arduinos, facilitating the smooth operation of our Smart Traffic Control System.

Original work:

- We have innovatively integrated affordable Arduino components and advanced wired communication protocols to create a bespoke system for real-time traffic management.
- These custom solutions not only address specific challenges in pedestrian safety and traffic flow but also highlight our project's distinctiveness and value in the realm of urban infrastructure management.

What worked

1. **Serial Communication (UART):**

- Seamless data transmission and reception.
- Ensured reliable communication across the system.

2. **Simplicity in Implementation:**

- Smooth and efficient implementation process.
- Allowed focus on critical aspects like sensor integration and algorithm development.

3. **Reliable Data Transfer:**

- Error-checking mechanisms ensured data integrity.
- Crucial for accurate communication in dynamic traffic scenarios.

4. **Compatibility:**

- Seamless integration within our system architecture.
- Enhanced overall system cohesion and functionality.

What doesn't work or what caused problems:

1. **Limited Range of UART**

- Maintaining reliable communication over longer distances was challenging.

2. **Interference and Noise**

- Susceptibility to interference and noise in high-electromagnetic environments caused disruptions.

3. **Complexity in Multi-Device Communication**

- Managing communication among multiple Arduinos introduced complexity and synchronization issues.

4. **Resource Utilization**

- High resource consumption in data-intensive applications impacted system performance.

Highlight process/team related roles

1. **Hardware Development:**

- Designed and implemented hardware components such as Arduino microcontrollers and sensors.
- Ensured compatibility and functionality of hardware elements.

2. **Software Development:**

- Developed software algorithms for traffic management and communication protocols.
- Conducted testing and optimization of software components.

3. **Communication and Integration:**

- Implemented communication protocols (e.g., UART) for inter-Arduino communication.
- Integrated hardware and software components into a cohesive system.

4. **Testing and Quality Assurance:**

- Conducted thorough testing procedures to identify and resolve issues.
- Ensured system reliability, performance, and adherence to project requirements.

5. **Documentation and Presentation:**

- Documented project progress, including design choices, challenges, and solutions.
- Prepared presentation materials and reports for showcasing the project.