Smart Parking System Using ESP32

Project Report

Submitted by:

Name: Mukund Vishwas Chavan

Course: Software for Embedded Systems

Student ID: 01011

Student E-mail: 2024ht01011@wilp.bits-pilani.ac.in

Table of Contents

1. **Introduction**
    - **1.1 Project Background**
    - **1.2 Objectives**
    - **1.3 Scope of the Project**
2. **Literature Review**
    - **2.1 Existing Parking Solutions**
    - **2.2 Limitations of Traditional Parking Systems**
3. **System Design**
    - **3.1 Hardware Components**
    - **3.2 Software Components**
    - **3.3 Circuit Diagram**
    - **3.4 Actual Setup**
4. **Implementation**
    - **4.1 Sensor Integration**
    - **4.2 Microcontroller Configuration**
    - **4.3 OLED Display Setup**
5. **Code Explanation**
    - **5.1 Initialization**
    - **5.2 Main Logic**
    - **5.3 User Interface**
6. **Testing and Results**
    - **6.1 Testing Methodology**
    - **6.2 Test Cases**
    - **6.3 Results**
7. **Conclusion and Future Work**
    - **7.1 Conclusion**
    - **7.2 Future Enhancements**
8. **References**
9. **Github Link**

1. Introduction

1.1 Project Background

Parking management is a significant challenge in urban areas due to the increasing number of vehicles and limited parking spaces. As urbanization continues to grow, the demand for efficient parking solutions becomes more critical. Traditional parking systems often rely on manual intervention, which can be time-consuming, inefficient, and prone to human error. These systems typically involve parking attendants or drivers manually searching for available spots, leading to increased traffic congestion, wasted time, and higher emissions.

The advent of smart technologies offers a promising solution to these challenges. By leveraging embedded systems, sensors, and actuators, it is possible to create automated parking management systems that enhance efficiency, reduce human intervention, and provide a better user experience. This project presents a smart parking system that automates the process of checking parking spot availability and controlling access to the parking area.

The proposed system utilizes an **ESP32 microcontroller** as the central processing unit. It integrates various sensors and actuators to monitor and manage parking spaces effectively. An **ultrasonic sensor** is employed to detect the arrival of a car at the entrance, ensuring that the system can respond promptly to incoming vehicles.

IR sensors are strategically placed to check the availability of individual parking spots. When a car occupies a spot, the corresponding IR sensor is triggered, indicating that the space is occupied. A **servo motor** controls a flap mechanism that grants or denies access based on the availability of parking spots, ensuring that only authorized vehicles can enter.

Additionally, a **0.96-inch OLED display** provides real-time status updates regarding the availability of parking spots. This display continuously refreshes to show whether parking is available and the number of open spots, offering clear and immediate information to users.

1.2 Objectives

- To create a system that detects the availability of parking spots using IR sensors.
- To utilize an ultrasonic sensor to confirm vehicle proximity.
- To provide feedback through an OLED display.
- To control a servo motor for an automated flap mechanism.

1.3 Scope of the Project

This project will focus on a two-parking spot system that detects vehicle presence and provides real-time information on parking availability. The system is designed for educational purposes and can be scaled or modified for larger implementations.

2. Literature Review

2.1 Existing Parking Solutions

Existing parking solutions range from manual management systems to automated systems integrated with mobile applications. While some solutions provide real-time updates, many still rely on visual inspections, leading to inefficiencies.

2.2 Limitations of Traditional Parking Systems

- **Manual Monitoring**: High labour costs and errors due to human oversight.
- **Inefficiency**: Lack of real-time data can lead to overcrowding or underutilization.
- **User Experience**: Limited feedback for users regarding availability.

3. System Design

3.1 Hardware Components

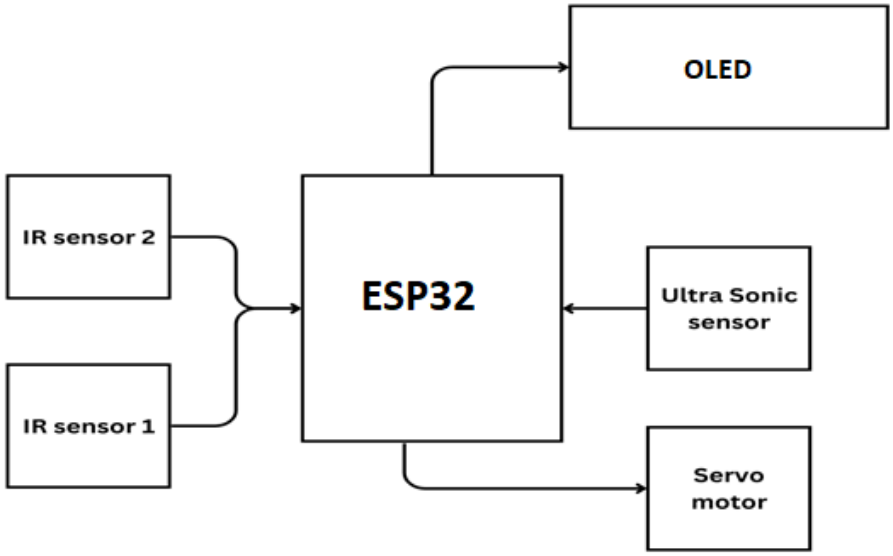
- **ESP32 Microcontroller**: The main controller for processing inputs and controlling outputs.
- **IR Sensors**: Two sensors to detect vehicle presence in parking spots.
- **Ultrasonic Sensor**: To measure the distance of vehicles approaching the parking area.
- **Servo Motor**: Controls the flap mechanism for granting access.
- **OLED Display**: Provides real-time information on parking status.

3.2 Software Components

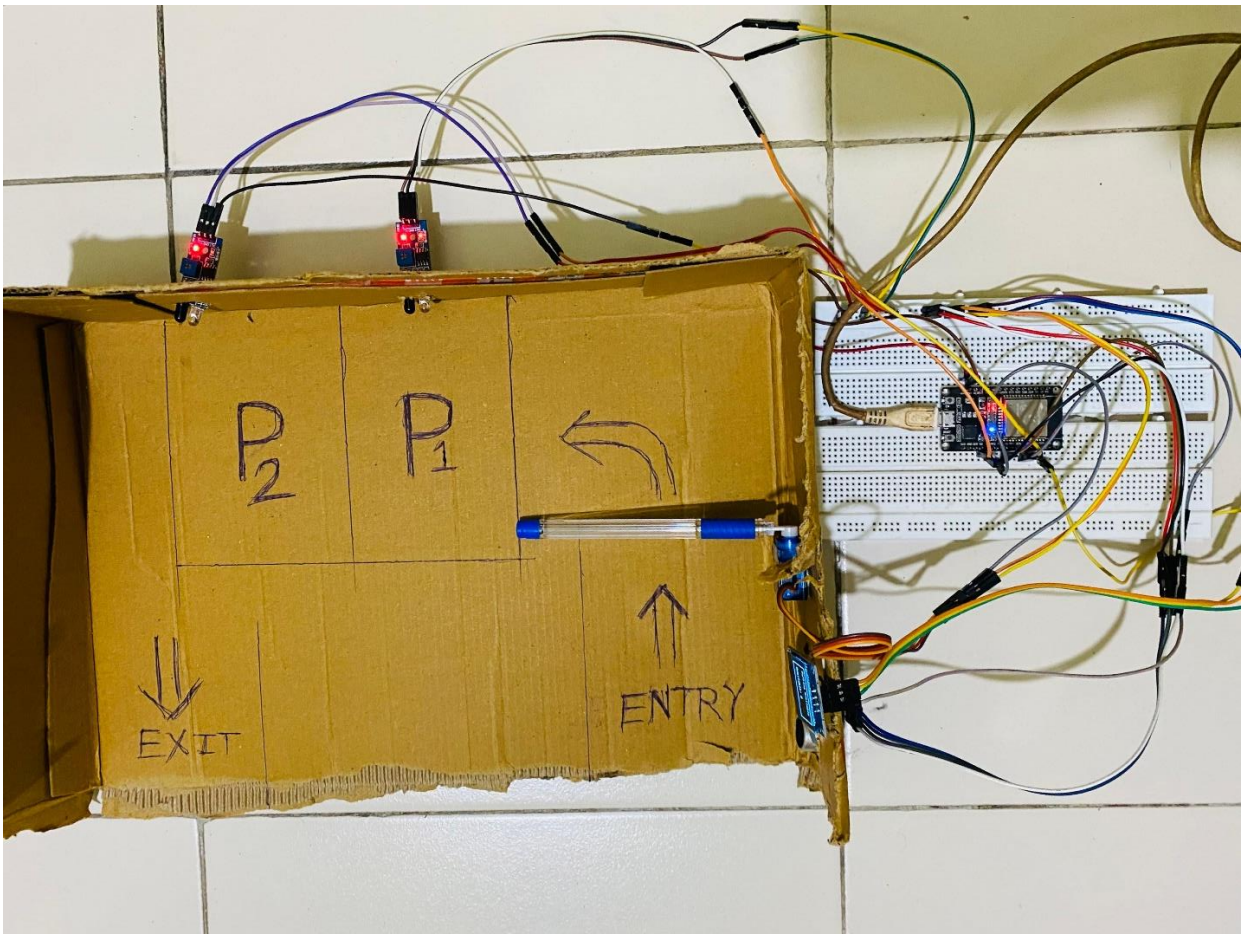
- **Arduino IDE**: Used for coding and uploading the firmware to the ESP32.
- **Adafruit Libraries**: For handling OLED display and GFX graphics.

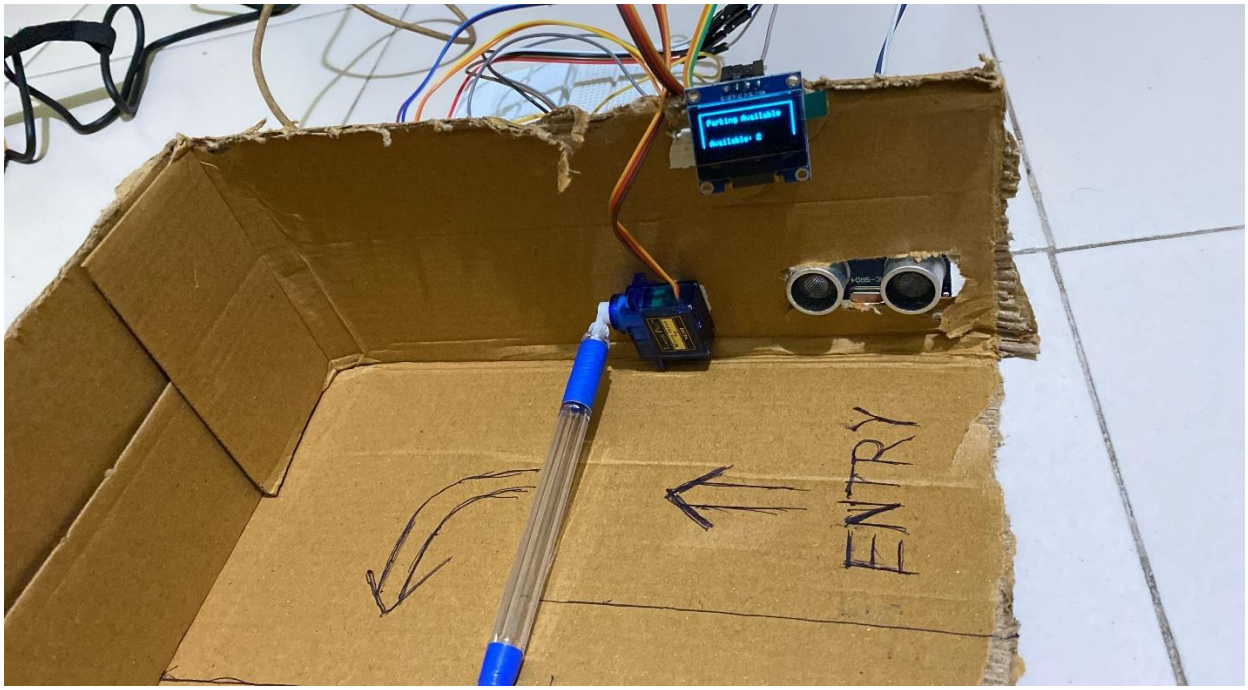
3.3 Circuit Diagram

A simple circuit diagram would illustrate how each component connects to the ESP32, showcasing the IR sensors, ultrasonic sensor, servo motor, and OLED connections.



3.4 Actual Setup





4. Implementation

4.1 Sensor Integration

The IR sensors are placed at designated parking spots to detect vehicle presence. The ultrasonic sensor is mounted at the entrance to monitor incoming vehicles.

4.2 Microcontroller Configuration

The ESP32 is configured to read inputs from the IR sensors and the ultrasonic sensor. It is programmed to control the servo motor and update the OLED display accordingly.

4.3 OLED Display Setup

The OLED display is connected via I2C to the ESP32, providing a clear interface for users to view parking status and availability.

5. Code Explanation

5.1 Initialization

The setup() function is critical for preparing the hardware components before the main program begins. Here's what it does:

- **Pin Configuration: The function sets the modes for each pin connected to the hardware:**
    - **IR Sensors:** Configured as inputs to detect the presence of vehicles in the parking spots.
    - **Ultrasonic Sensor:** The trigger pin is set as an output to send signals, while the echo pin is configured as an input to receive distance measurements.
    - **Servo Motor:** Attached to a specific GPIO pin for controlling the flap mechanism.
- **OLED Display Initialization:** The display is initialized and cleared to ensure that it's ready to show information. The text size and color settings are also defined here to ensure readability.
- **Initial Servo Position:** The servo motor is set to a starting position (flap closed) to begin in a safe state.

5.2 Main Logic

The loop() function runs continuously and implements the main functionality of the system:

- **Sensor Reading**: It reads the states of the two IR sensors to determine if the corresponding parking spots are occupied (when the sensor is covered) or available (when the sensor is uncovered). The ultrasonic sensor measures the distance to detect if a vehicle is approaching the parking area.
- **Parking Availability Calculation:** Based on the readings from the IR sensors, the code calculates how many parking spots are available. If either sensor detects an open spot, the count of available spots increases.
- **Flap Control:** If a vehicle is detected by the ultrasonic sensor within a predefined distance (e.g., less than 10 cm) and at least one parking spot is available, the servo motor is activated to open the flap, allowing the vehicle to enter. After a short delay, the flap is closed again.
- **OLED Display Update:** The display is cleared and updated with the latest information regarding parking availability. It shows whether parking spots are available or if the lot is full, providing clear feedback to users.

5.3 User Interface

The user interface on the OLED display is designed for simplicity and clarity:

- **Status Indicators: The OLED continuously shows the status of parking spots. It indicates "Parking Available" if at least one spot is open and "No Parking Spots" if both are occupied.**
- **Flap Status: Although not explicitly shown in the code snippet, the interface can be extended to display the flap status (open or closed) to inform users whether they can enter.**
- **Real-Time Updates: The display is refreshed every half second to ensure that users receive timely updates as vehicles enter or leave the parking area.**

6. Testing and Results

6.1 Testing Methodology

The system was tested under various scenarios to ensure accurate detection and reliable operation. Tests included checking the response of the IR sensors, ultrasonic distance measurements, and the servo motor's operation.

6.2 Test Cases

- **Case 1**: Both parking spots available → Expected: "Parking Available"
- **Case 2**: One parking spot occupied → Expected: "Available: 1"
- **Case 3**: Both spots occupied → Expected: "No Parking Spots"

6.3 Results

The system successfully identified parking availability in real-time and controlled the flap as intended. The OLED display accurately reflected the status of parking spots.

7. Conclusion and Future Work

7.1 Conclusion

The Smart Parking System effectively utilizes sensors and a microcontroller to provide real-time parking availability feedback. This system demonstrates a practical approach to improving urban parking management.

7.2 Future Enhancements

Future work on the Smart Parking System could include:

Mobile Application Integration: Developing a mobile app for remote monitoring of parking availability. Users could check available spots in real-time, receive notifications about spot availability, and access directions to the parking location.

Reservation System: Implementing a reservation system that allows users to book parking spots in advance through the mobile app. This feature would help ensure that users have a guaranteed space upon arrival, improving convenience.

Integration with Smart City Infrastructure: Linking the Smart Parking System with broader smart city initiatives to improve urban traffic management. For example, the system could communicate with traffic lights or navigation apps to direct vehicles to available parking spots.

Payment Integration: Adding a payment system that allows users to pay for parking through the mobile app. This could streamline the payment process and reduce the need for physical cash or tickets.

Sustainability Features: Introducing features to promote eco-friendly practices, such as prioritizing electric vehicle charging stations or offering discounts for carpooling.

User Feedback System: Creating a feedback mechanism within the mobile app where users can report issues or suggest improvements, helping to continuously enhance the system based on user experiences.

Integration with Other Transportation Modes: Exploring options for integrating the parking system with public transportation schedules, allowing users to plan their journeys more effectively.

8. References

- Arduino Documentation
- Adafruit SSD1306 Library Documentation
- Previous studies on smart parking systems and sensor technology.

9. Github Link

Please click here for project and video