**PHYSICS PROJECT REPORT**

ARDIUNO SMART CAR PARKING SYSTEM

**PRESENTORS:**

* Nimra Kamran
* M. Sabayel
* Ahmad Ali
* Khansa bibi
* Hassan Khan
* Pakeeza Naeem
* Abbas Munir

**REPORT:**

INTRODUCTION:

The Arduino-based car parking system is an innovative project that integrates physics principles with modern technology to automate and enhance parking management. This system employs sensors and microcontrollers to detect and manage the availability of parking spaces, providing real-time data to users and optimizing parking efficiency.

PURPOSE:

The primary purpose of this project is to develop an efficient, reliable, and user-friendly car parking system. By automating the process, we aim to reduce human error, minimize time spent searching for parking spaces, and streamline traffic flow within parking facilities. Additionally, this system can be extended to smart city applications, contributing to better urban mobility and reduced carbon emissions.

**COMPONENTS:**

* **Arduino UNO**
* **2 IR sensor**
* **Servo motor**
* **Breadboard**
* **Jumper wire**
* **12C module**
* **7 Segment LCD Display**
* **Battery Holder**
* **Battery**

**Working:**

1. **Detection**: The two IR sensors are placed at the entrance and exit of the parking lot. These sensors detect the presence of a vehicle as it enters or exits the parking space. The sensors work by emitting infrared light and measuring the reflection from nearby objects.
2. **Signal Processing**: When a vehicle is detected by the IR sensors, the signal is sent to the Arduino UNO microcontroller. The Arduino processes this signal to determine whether a vehicle is entering or leaving the parking space.
3. **Parking Space Management**:
   * **Entry**: When a vehicle enters the parking lot, the entry IR sensor sends a signal to the Arduino. The Arduino then updates the count of available parking spaces. If a parking space is available, the Arduino triggers the servo motor to open the gate, allowing the vehicle to enter.
   * **Exit**: When a vehicle exits the parking lot, the exit IR sensor sends a signal to the Arduino. The Arduino updates the count of occupied parking spaces and triggers the servo motor to open the gate, allowing the vehicle to leave.
4. **Display**: The 7-segment LCD display, connected via the I2C module, shows the number of available parking spaces. The Arduino updates the display in real-time based on the signals received from the IR sensors. For example, if the parking lot has 10 spaces and a vehicle enters, the display will show 9 available spaces.
5. **Power Supply**: The entire system is powered by batteries placed in the battery holder. The power supply ensures that the Arduino, sensors, servo motor, and display are all operational.

**Component Connections**:

* **IR Sensors**: The IR sensors are connected to the digital input pins of the Arduino to detect vehicle entry and exit.
* **Servo Motor**: The servo motor is connected to one of the PWM (Pulse Width Modulation) pins on the Arduino for controlling the gate mechanism.
* **7-Segment LCD Display**: The 7-segment LCD display is connected to the Arduino via the I2C module, which simplifies the wiring and communication between the display and Arduino.
* **Breadboard and Jumper Wires**: The breadboard and jumper wires are used to make connections between the components and the Arduino, ensuring a neat and organized setup.
* **Battery**: The battery provides power to the entire system, and the battery holder ensures a secure connection.

**Operation Flow**:

1. **Initialization**: The Arduino is powered on, and the initial number of available parking spaces is displayed on the LCD.
2. **Vehicle Entry**: As a vehicle approaches the entrance, the IR sensor detects it and sends a signal to the Arduino. The Arduino processes the signal, updates the display, and controls the servo motor to open the gate.
3. **Parking Space Update**: The display is updated to show the reduced number of available spaces.
4. **Vehicle Exit**: As a vehicle exits, the exit IR sensor detects it and sends a signal to the Arduino. The Arduino processes the signal, updates the display, and controls the servo motor to open the gate.
5. **Parking Space Update**: The display is updated to show the increased number of available spaces.

FUTURE OBJECTIVES

1. **Scalability**: Expand the system to accommodate larger parking facilities or integrate it with city-wide smart parking networks.
2. **Mobile Integration**: Develop a mobile application that allows users to reserve parking spaces in advance and receive real-time updates on space availability.
3. **Energy Efficiency**: Optimize the system to reduce energy consumption through the use of low-power components and sustainable energy sources.
4. **Data Analytics**: Incorporate data analytics to analyze parking patterns and optimize space utilization.

