**Arduino Car Parking**

**System**

Dean Michael Villacorta

Karl Louie Seda

Patricia Ara C. Marquez

Glecelyn O. Peroso

Jamal Suralta

**Introduction**

**Project Context**

An Arduino car parking system is a project that involves using an Arduino microcontroller to create a system that helps drivers park their cars. The project typically involves installing sensors that detect the presence of a car in a parking space, and using the Arduino to display information about available parking spaces on an LCD display.

The system can be designed to use infrared sensors to detect the presence of a car in a parking space. The sensors are connected to the Arduino, which processes the data and sends the information to the display.

The LCD display can be mounted in a visible location, such as near the entrance of a parking garage, to provide real-time information about available parking spaces.

The Arduino car parking system can be used in a variety of settings, including parking garages, shopping malls, and airports. It can help drivers save time and reduce frustration by providing accurate information about available parking spaces in real-time.

**Purpose and Description of the project**

The purpose of an Arduino car parking system is to provide drivers with real-time information about available parking spaces, thereby reducing the time and frustration associated with finding a parking spot. The system uses sensors to detect the presence of a car in a parking space, and displays the information on an LCD display.

**Objectives of the project**

The objective of the project is, it can help drivers save time and reduce frustration by providing accurate information about available parking spaces in real-time and create a more efficient and user-friendly parking experience that benefits both drivers and parking operators.

­­­­­­

**Technical Background**

**Technicality of the Project**

The technicality of the project is it involves the programming of the Arduino board to read and interpret the signals from the sensor and servo motor.

**Details of the technologies to be used**

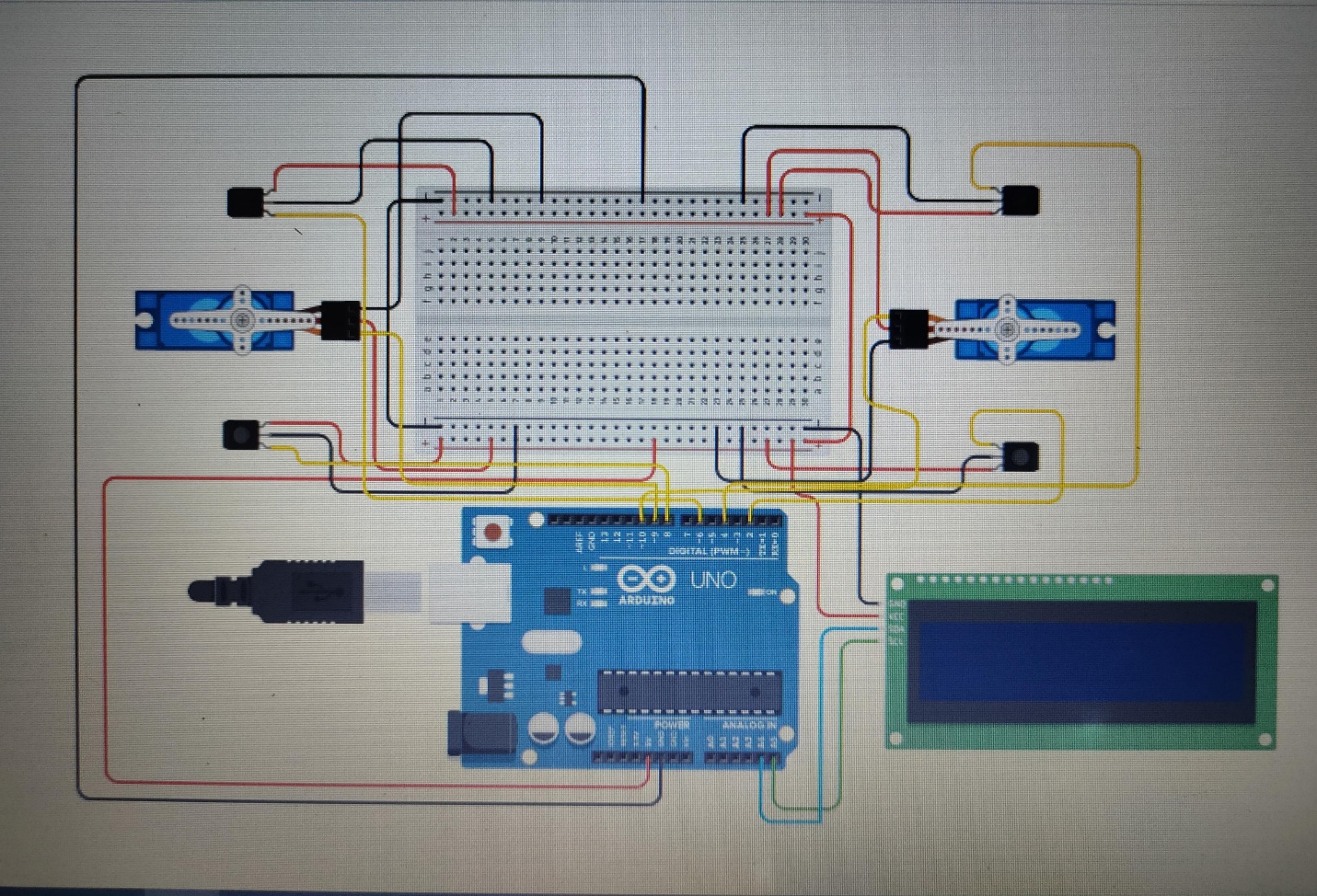
The Arduino car parking system typically consists of the following components:

1. Infrared Sensor: The sensors are used to detect the presence of a car in a parking space. Ultrasonic sensors or infrared sensors are commonly used for this purpose.
2. Arduino Uno: The Arduino microcontroller is the brain of the system. It receives input from the sensors and processes the data to determine the availability of parking spaces.
3. LCD Display: The display can be an LCD display. The LCD display is typically mounted in a visible location, such as near the entrance of a parking garage, and provides real-time information about available parking spaces.
4. Power supply: The system requires a power supply to operate.
5. Jumper wire: Used for making connections between the sensor, servo motor, and lcd in display to Arduino uno.

**How the project will work**

The project uses infrared sensors to detect the presence of a car in a parking entrance and then communicates the information to a central control unit, which can then display the availability of the parking space to drivers.

**Circuit diagram of the project**



**Methodology**

**Data and Process Modeling**

**Context Diagram** **& Data Flow Diagram**

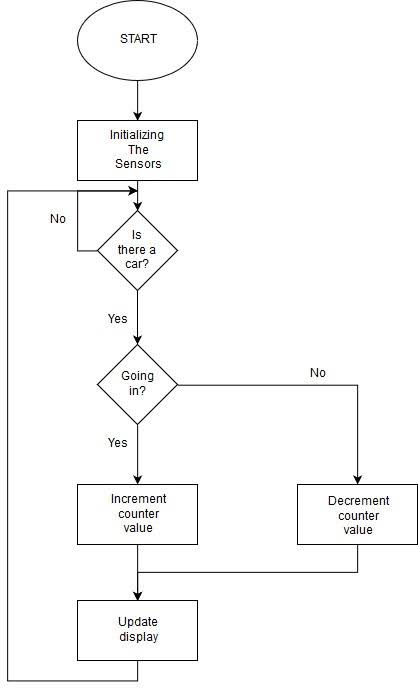
Parking Slot

Management

Parking Management

Parking Space

Management

**System Flow Chart & Program Flowchart**

**System Architecture**

Arduino Uno

IR Sensor In

IR Sensor Out

Servo Motor

Servo Motor

IR Sensor Out

IR Sensor In

LCD Display I2C

**Network Topology**

The Arduino Car Parking system proposed in [5] adopts a single-hop sensor network with a star topology, where a plurality of sensor nodes for detecting the availability of parking spaces communicate with a sink node located at the center of the network.

**Parts and Equipment Needed**

To create an Arduino car parking system, you will need several parts and equipment. Here is a list of the basic components required for such a project:

* Arduino Uno
* LCD Display 16x2
* Jumper Wires
* Servo Motor
* Infrared Sensor
* Breadboard

If you want to protect your system and give it a finished look, you can use an enclosure to house all the components.

**Software Specifications**

The software specifications of an Arduino car parking system involve programming the Arduino board to control and monitor the various components of the system.

**Installation Process**

The installation process of an Arduino car parking system involves setting up the hardware components, connecting them properly, and uploading the software code to the Arduino board. Here’s a step-by-step guide:

* Gather all the required components: Collect the Arduino board, infrared sensor sensors, servo motor, breadboard, jumper wires, LCD display 16x2, and any optional components you plan to use.
* Design and assemble the circuit: Create a circuit diagram or refer to the wiring diagram provided by the sensor and motor manufacturers. Connect the Arduino board, infrared sensors, servo motor, LCD Display and other components on the breadboard. Use jumper wires to establish the necessary connections. Pay attention to correct pin assignments and ensure proper power and ground connections.
* Power the system: Connect the power supply to the Arduino board, making sure the voltage and current ratings are appropriate for your components. Double-check the polarity to avoid damaging the components.
* Install the sensors and motor: Position the infrared sensors at suitable locations to detect incoming and parked vehicles. Mount the servo motor to control the parking barrier or gate.
* Upload the code: Launch the Arduino IDE on your computer and connect the Arduino board via USB. Open the Arduino code file for the car parking system that you have written or obtained. Compile the code to check for any errors. If there are no errors, upload the code to the Arduino board.
* Test the system: After uploading the code, disconnect the Arduino from the computer and power it using the external power supply or battery pack. Place a vehicle in the detection zones of the sensors and verify that the system functions as expected. Observe the LED indicators and the movement of the servo motor to confirm correct operation.
* Fine-tune and troubleshoot: If you encounter any issues during testing, check the connections, review the code for errors, and refer to the datasheets or documentation of the components. Make necessary adjustments to the code or hardware configuration to resolve any problems.
* Enclosure and final installation: Once you are satisfied with the system’s functionality, consider placing the components inside an enclosure to protect them and give the setup a finished appearance. Mount the enclosure in the desired location for the car parking system, ensuring proper sensor alignment and accessibility.

**Building the Circuits**

* Gather the necessary components: Prepare the Arduino board, infrared sensors, servo motor, breadboard or PCB, jumper wires, LCD, and any other required components.
* Connect the Arduino board: Place the Arduino board on the breadboard or PCB. Connect the power and ground pins (VCC and GND) of the Arduino to the corresponding rails on the breadboard or PCB.
* Connect the infrared sensors:

Incoming Vehicle Sensor: Connect the VCC pin of the incoming vehicle sensor to the 5V rail on the breadboard or PCB. Connect the GND pin to the ground rail. Connect the out pins of the sensor to any digital pins of the Arduino board.

Outgoing Vehicle Sensor: Connect the VCC pin of the outgoing vehicle sensor to the 5V rail. Connect the GND pin to the ground rail. Connect the out pins of the sensor to different digital pins of the Arduino board.

* Connect the servo motor:

Connect the VCC (usually red) and GND (usually brown or black) wires of the servo motor to the 5V and ground rails, respectively.

Connect the control wire (usually yellow or orange) of the servo motor to a digital pin of the Arduino board.

* Connect additional components (if applicable):

If you are using an LCD display, connect its power, ground, and data pins to the corresponding pins on the Arduino board.

If you are using a keypad or other input devices, connect them to the appropriate pins on the Arduino.

* Double-check connections: Ensure that all connections are properly made, and there are no loose wires or accidental short circuits.
* Power the system: Connect the power supply (battery pack or regulated power adapter) to the Arduino’s power jack or Vin pin.

**Testing**

**System testing**

System testing is an important step to ensure the proper functioning of an Arduino car parking system. Here’s a guide on how to test our Arduino Car Parking System:

* Power up the system: Connect the power supply to the Arduino board and ensure it is properly powered on.
* Verify sensor readings: Test the infrared sensors by placing a vehicle in front of each sensor. Verify that the sensors accurately detect the presence of the vehicle and provide distance measurements.
* Check LCD display: Observe the LCD display connected to the Arduino board. Ensure that the LCD correctly indicate the parking space status based on the sensor readings.
* Test servo motor control: Trigger actions with the servo motor based on the sensor inputs.

Users guide

User Guide: Arduino Car Parking System

Congratulations on setting up your Arduino car parking system! This user guide will help you understand how to operate and utilize the system effectively. Please follow these instructions to ensure a smooth user experience:

* **Powering On the System:**

1. Connect the power supply to the Arduino board and ensure it is properly powered on.
2. Wait for the system to initialize. The LCD indicators may light up briefly during startup.

* **Parking Space Status:**

1. Observe the LCD indicators connected to the Arduino board.
2. The LCD will show the available vacant parking space.

* **Parking a Vehicle:**

1. When entering the parking area, drive towards the entrance.
2. When incoming infrared sensor will detect your vehicle and trigger the parking barrier to remain raised, allowing you to enter.

* **Exiting the Parking Area:**

1. When leaving the parking area, drive towards the exit.
2. When the parked vehicle exit the infrared sensor will detect your vehicle and keep the parking barrier lowered, allowing you to exit.

* **Safety Precautions:**

1. Always follow traffic rules and exercise caution while driving within the parking area.
2. Maintain a safe distance from the parking barrier and other vehicles to avoid accidents.

* **System Maintenance:**

1. Regularly inspect the infrared sensors, servo motor, and other components for any signs of damage or malfunction.
2. If you notice any issues or abnormal behavior, refer to the troubleshooting section or seek technical support.

* **Troubleshooting:**

1. If the parking barrier does not raise or lower correctly:
2. Check the connections of the servo motor and ensure they are securely attached.
3. Verify that the sensor inputs and logic in the code are properly programmed.
4. If the LCD display do not display the correct parking space status:

* Check the connections of the LCD
* Review the code to ensure the correct LED pins are defined and used.
* **System Shutdown:**

1. To power off the system, disconnect the power supply from the Arduino board.