Multilevel Car Parking System using Arduino

*Abstract*— The project aims to address urban parking management issues by designing and implementing a multilevel parking system using microprocessor and embedded systems and digital logic components like logic gates, servo motors, sensors, Arduino boards, and a 7-segment display. The system aims to maximize available parking slots, lessen overcrowding, and enhance the overall urban living experience. The integration of microprocessor and advanced digital logic technologies automates the process, minimizing human intervention. The system's reliability is ensured by precise movement of the parking mechanism, while sensors enable real-time detection and management. The Arduino board serves as the central control unit, while a 7-segment display provides clear feedback on parking status. This project contributes to smart urban infrastructure development.

Keywords—parking management, arduino, multilevel, traffic control, embedded system

# Introduction

## Background Information

Parking management is a great hassle nowadays in urban people’s life. It causes heavy traffic difficulty and an intend to causing illegal parking on streets. This all mismanagements result in great inconvenience, time waste in economical and daily life works. Parking management systems are now a crucial component of the entire parking process for parking operators and consumers thanks to the development of parking technology. These days, parking management is complicated without the right method and proper execution. Using parking management systems to regulate the movement of cars in a parking lot may be rather practical and adaptable. Through this project, a multilevel parking management system can be constructed which is necessary for parking management, especially in busy cities.

## Project Objectives

The proposal involves a multi-level parking structure This project's main objective is to construct a system that, in order to prevent excessive vehicle entry jams and manage car parking safety, would automatically respond to user requests for space availability by opening the gate based on which level has available parking space. This project is intended to create spaces for buildings with multi levels which is one of the highlights of this project.

## Brief Outline of this Project

This system integrates into urban infrastructure, providing convenience and flexibility for both operators and users. They enable dynamic monitoring and control of parking spaces, utilizing real-time data and analytics to optimize resource utilization. Users benefit from a user-friendly interface, streamlining the process of finding and securing parking spaces. These systems also reduce traffic congestion and environmental effect by reducing the amount of time spent looking for parking slots. Their adaptability allows for scalability, making them suitable for diverse parking environments. The relationship between advanced parking technologies and effective management systems represents a paradigm shift in urban planning, promoting a more organized, efficient, and user-centric approach to parking in modern cities.

# Literature Review

The requirement for parking spots and the scarcity of available land are the two considerations that motivated the notion of the automated parking system. In 1905, the Garage Rue de Pontius in Paris, France, installed the first Automated Parking System (APS) [1]. Since the 1970s, more technologically sophisticated APS has been installed across Europe, Asia, and Central America. In Japan, the paternoster APS was being used to construct almost 40,000 parking spots per year in the beginning of the 1990s. In Japan, there are thought to be 1.6 million APS parking places as of 2012. Due to the growing lack of accessible urban space for construction, the increase in the number of automobiles on the road, and other quality-of-life concerns, there is increased interest in APS as an alternative to parking lots, on-street parking, and multi-story garages [1][2].

# Methodology

## Introduction

In this section, a thorough description is provided of the parts used in the experiment setup, including the Arduino UNO, required integrated circuits, infrared sensor, seven-segment display, servo motor, and jumper wires on a breadboard. The components of the experimental setup are connected to create an automatic parking system that can be controlled by programming on an Arduino UNO.

## Working Principle of the Proposed Project

This project's operating concept comprises using an Arduino UNO microcontroller and an identification sensor to govern parking. This technological method coordinates the counting and automated gate opening procedure, continuously monitoring the inbound and leaving car surveillance. This ensures the best possible conditions for either fully occupied or empty parking spaces, improving accuracy and supporting the long-term growth of the multi-level parking system.

### Process of Work

In this system, two sensors at the entry gate are connected directly with the Arduino UNO board. Between these two sensors, a servo motor is there for the work of entrance gate opening and closing. The 7-segment display is connected with the Arduino UNO. The two ICs (NOT gate and NAND gate) are also connected with both Arduino and all the other sensors of each parking slot for input references.

First of all, this parking system is built in such a way that when a car comes in front of the gate, the sensor at the front counts the car and sends signal to Arduino. If there is any empty space then the gate will open and the appearance of a car in any empty slot will be counted and will be displayed in the7-segment display the current empty spots specifying floor number. When all the slots are full and there’s no empty slots, the entrance gate won’t be open anyways even if the sensor detects car appearance.

Similarly, if a car gets out of the parking space the entrance gate’s sensor will count and update the signal and the display will be updated showing a space has been empty. That’s how this entire system works.

## Description of the Components

The components used in the system are widely available, easy, and cheap to implement.

### 3-input NAND Gate

The 3-input NAND gate is a digital logic gate with three inputs and one output. This component is essential for building complex logical circuits and is often used in conjunction with other gates [].



1. 7410 IC – Triple 3 input NAND Gate [3].

### NOT Gate

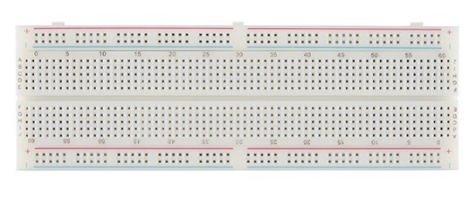
The NOT gate, also known as an inverter, is a basic digital logic gate with a single input and one output. It produces an output that is the logical complement of its input. NOT gates play a crucial role in logical operations and signal inversion [].



1. 7404 IC NOT GATE [4].

### Breadboard

A breadboard is a fundamental prototyping tool in electronics. It consists of a grid of interconnected sockets, allowing electronic components to be easily connected without the need for soldering. Breadboards facilitate the creation and testing of circuits by providing a convenient and reusable platform for assembling various components.



1. Breadboard [5].

### 16x2 LCD Display

A 16x2 LCD (Liquid Crystal Display) is one kind of display that can show up to 16 characters per line for 2 lines and is often used in electronic projects. In other words, the "16x2" indicates the display size, featuring 16 columns and 2 rows of characters. This display is capable of showing alphanumeric characters, making it suitable for presenting information and data in projects.

### 



1. LCD Display (16×2) [6].

### Servo Motor

A servo motor is a type of rotary actuator that is used to precisely regulate an angle. It is made up of a feedback control system, a gearbox, and a motor. Servo motors are commonly used in projects where accurate and controlled rotation is required, such as in robotics, automation, and, in this context, for controlling mechanical movements.



1. Servo motor [7].

### Jumper Wires

Jumper wires are essential for creating electrical connections on a breadboard or between various components in a circuit. These wires are flexible and have connectors at both ends, enabling easy and temporary connections. Jumper wires are instrumental in establishing the electrical pathways needed to complete a circuit and transfer signals between components.



1. Jumper wires [8].

### IR Sensor

An IR (Infrared) sensor is a device that can detect infrared radiation. In electronics, IR sensors are often used to detect the presence or proximity of objects. They work by emitting infrared light and measuring the reflection or absence of reflection. IR sensors find applications in various projects, including proximity detection, object counting, and obstacle avoidance.



1. IR sensors [9].

### Arduino UNO Board

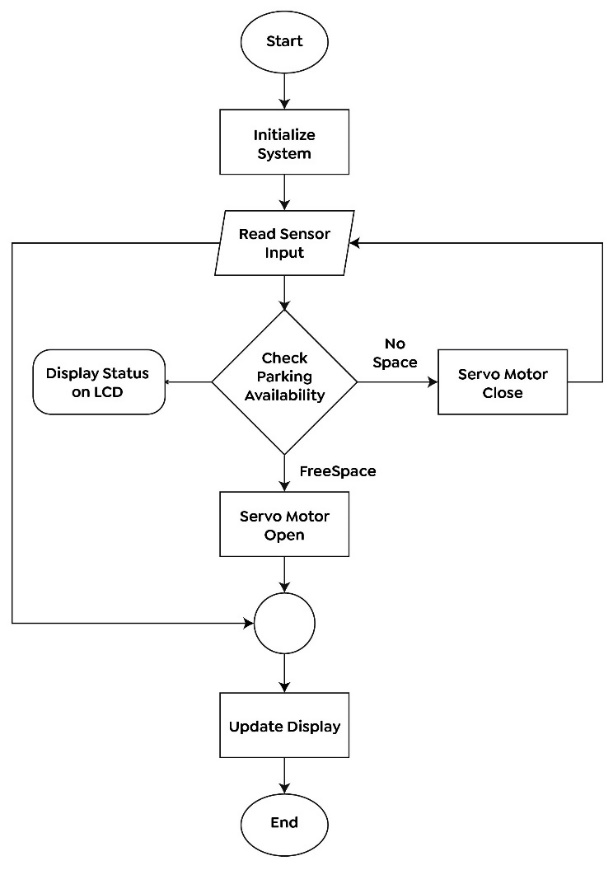
The Arduino UNO board is a widely acknowledged and popularly used microcontroller board which functions as the mastermind of many electronic projects. It is based on the ATmega328P microcontroller and comes equipped with digital and analog input/output pins, onboard voltage regulators, USB interface, and other essential components. The Arduino UNO is programmable using the Arduino IDE (Integrated Development Environment) and is widely used for prototyping and creating interactive electronic projects. It acts as the central control unit, executing the programmed instructions and coordinating the operation of other connected components in a project.



1. Arduino UNO board [10].

## Operation Flowchart

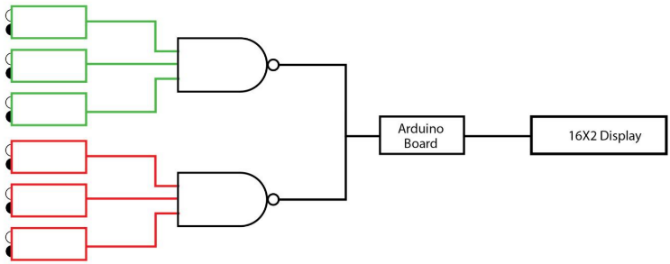
In the flowchart, it can be seen that after the starting of the flowchart, it is needed to initialize system by setting up the components including the Arduino UNO board, breadboard, and various sensors. Then sensor inputs are used to read the presence or absence of vehicles in parking spaces. Then the processing of the sensor inputs happens to determine the availability of parking spaces on each level. After that, display of real-time information regarding available parking places is done via the 16x2 LCD display. Then read and process any user input to decide if there's no parking space, then the servo motor remains closed and the loop of sensor reading input continues. If there's free space available, servo motor opens. After the servo motor control, reflection of any changes in the parking status is updated on the LCD display. The loop continues with updating and monitoring every time. Here's the end of flowchart.



1. Flowchart of the operation to be executed.

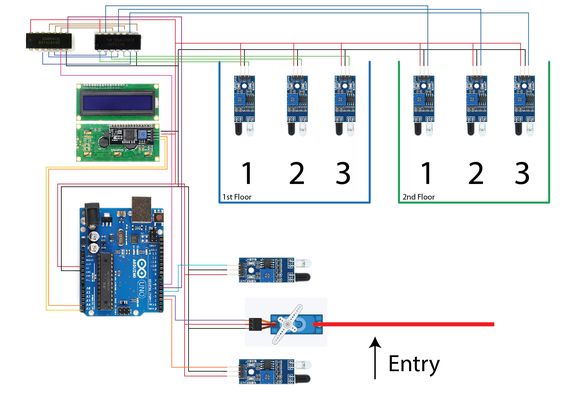
## Experimantal Setup

In figure no.10, it is a block diagram of the entire project with the building blocks with only components specified. It shows that the sensors of input are connected to ICs from which input signal and output results are prepared from the Arduino UNO board and displayed in 16\*2 display.



1. Block diagram of an arduino based automated parking system.

In figure 11, this exact diagram is expressed with proper visuality of the components that can help to build this system as explained earlier in the work process. It is clear that this is a multilevel system with 2 floors in this project.

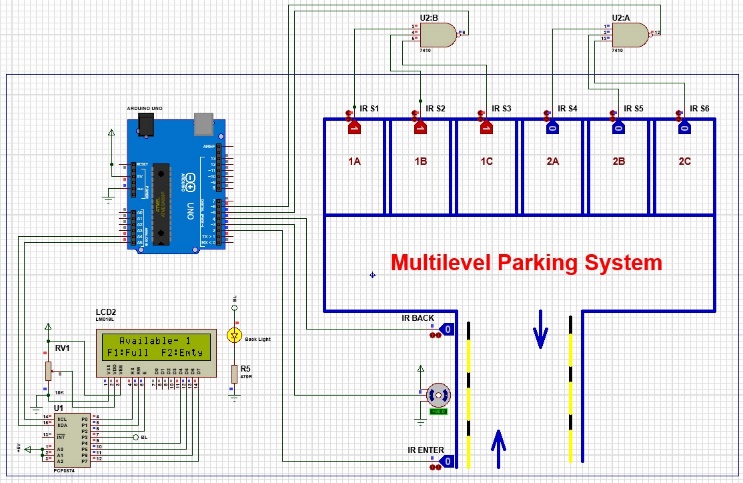


1. Circuit diagram of an arduino based automated parking system.

# Results And Discussions

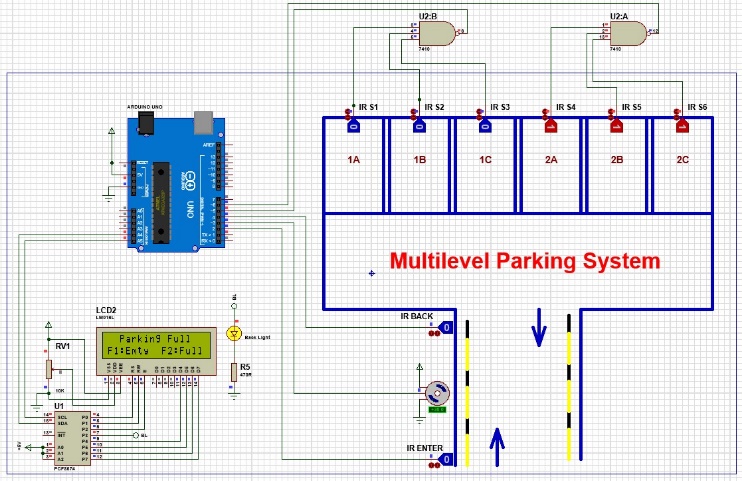
## Simulation Analysis

Simulation when the first floor is full:

****

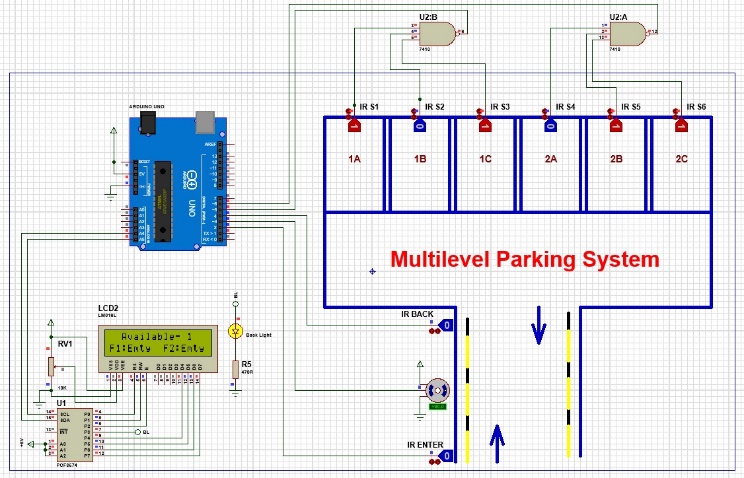
1. Simulated result when the first floor is full.

Simulation when the second floor is full:

****

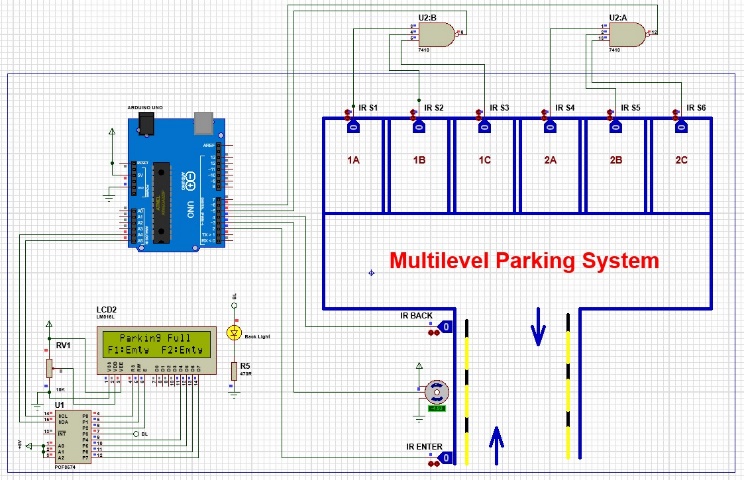
1. Simulated result when the second floor is full.

Simulation when there are 2 empty slots in 2 floors:

****

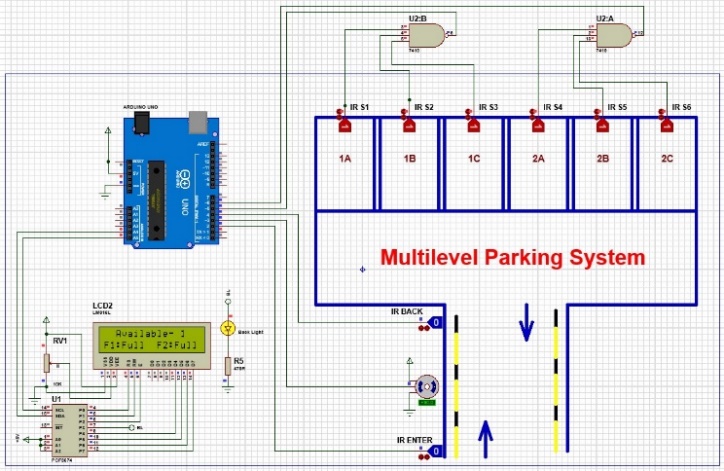
1. Simulated result when one slot of each floor is empty.

Simulation when all parking slots are empty:

****

1. Simulated result when all parking slots are empty.

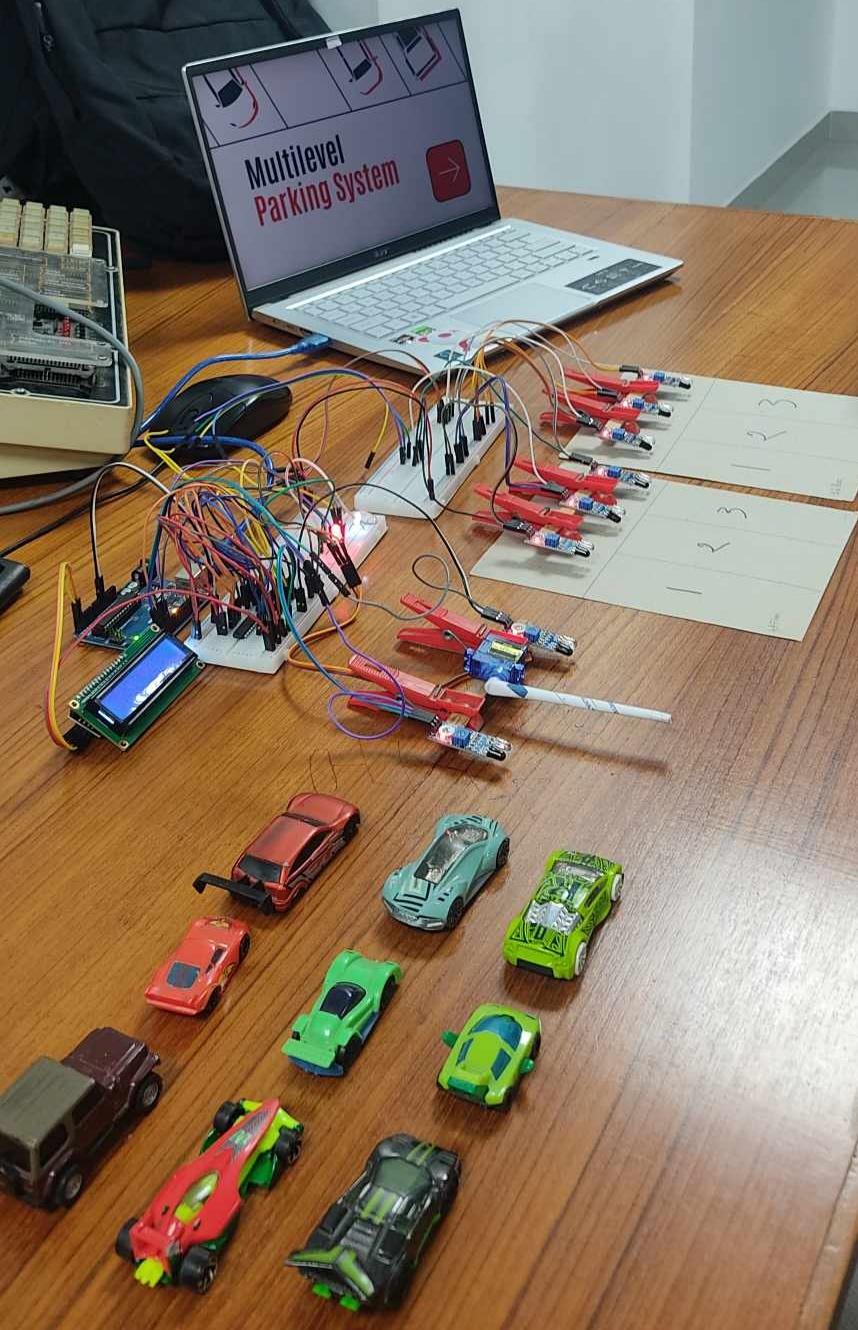
Simulation when all parking slots are full:

****

1. Simulated result when all parking slots are full.

## Experimantal Results

This paper explored the idea of a multilevel parking system that employs sensors to identify whether or not a spot is available for free. In this lab, we have used Proteus software for the simulations in this work. To get started, it was needed to learn about all the components and their features. In practical, we have implemented the whole system and tested its ability. The components were then successfully and error-free assembled on the Proteus bread board. Finally, both experimental and simulated observations showed the functionality as expected.



1. Overall experimental setup.



1. Setup with all the components implemented.

## Limitations of the Project

There were various types of sensors, their applications, advantages and limitations were also discussed. Such that, here, IR sensors were used which are low power consumption, high sensitivity and high dynamic range, extremely compact and inexpensive.

In this parking system, if any IR sensor has defects, accurate vehicle presence won't be detected. For example, this system could have been built with only AND gate but for the problem with our sensor, it could not give the right outputs for AND gate, so a NOT gate and a NAND gate had to be used for this problem.

Similarly, if the display has any defect, it will not show anything of available spaces. Here in every parking space sensor was used. But in a bigger parking system, this much sensors along with other components are costly. In case, any component gets damaged, we have to change it immediately.

It can be Arduino Uno board or logic gates. Also, for more floors, these much of sensors can be extremely difficult to handle when there is any need of detecting any problematic sensor at any parking spot.

# Conclusion And Future Endeavors

As the conclusion of this report, the multilevel parking system project using microprocessor and digital logic component, sensor, Arduino board provides an effective and reliable solution for managing parking spaces. The system accurately detects vehicle entry and parking availability, guides driver to empty spot and incorporates features light display and LEDs. Future advancement to this system can make it more easy and secure to use.

##### References