University of Windsor

Computational Methods and Modeling for Engineering Applications

(GENG8030-4-R-2023F)

Final Report on Smart Parking System

Submitted by: Group24

Name	Student Number
Utkarsh Dwivedi	110135891
Nawazhusain Shaikh	110125044
Minal Patil	110124010

Submitted to:

Dr. Mohammad Sedigh Toulabi

Table of Contents

List of Figures

1. Project Objective:

One of the biggest challenges people have to deal with nowadays is finding parking spaces. Smart parking systems are one potential solution to this problem. In order to effectively allocate parking spaces and provide customers with a pleasant and secure parking experience, the use of advanced parking technologies is necessary. The primary objective of the initiative is to develop and build a system that uses an arduino circuit and MATLAB code.

2. Project Description:

The goal of our project is to develop an advanced parking system using Arduino. The tasks for this project shall be to use an Arduino chip and write MATLAB code in line with the Project's specifications. The design of a system which makes use of an LCD display to notify cars when parking space is available shall be one of the main objectives. If there are no seats available, a message will appear asking them to return later, such as "plz come later." However, if there are spaces in the list, a greeting like "welcome" will appear.

When you go into the parking lot, the barrier gate is shut and there's a Red Traffic Sign flashing. To initiate an entry procedure, a customer must push the enter button. Later, a green traffic signal will come on to indicate that the barrier arm is going to open. When you park your car, the available parking spaces will drop one at a time, the light switches from red to green and the barrier arm closes. Likewise, when you pull out of the parking spot and hit the Exit button, it causes a barrier arm to be opened and an oncoming green light to flash. A further parking space will be available at the same time.

This project will use an Arduino microcontroller, together with a number of devices, in order to replicate the smart parking system. There are 13 parking spaces in the allocated area and they shall be equipped with essential features such as entry and exit buttons, barrier gates, an LCD screen showing information on how many places are available or vacant, and indicator lights indicating whether a place is open. A Programmable Microcontroller will serve as a main control unit of the parking system, coordinating all its functions.

3. Components:

3.1. Arduino UNO:

The Arduino UNO is a fundamental board in Arduino's lineup, serving as the core microcontroller overseeing parking system operations. Comprising shields, diverse circuits, and an array of digital and analog Input/ Output pins, this board boasts 14 digital pins, a USB port, a power jack, and an ICSP (In-Circuit Serial Programming) header. Moreover, it provides 6 analog pin inputs. Programming for the UNO is facilitated through an Integrated Development Environment [1].

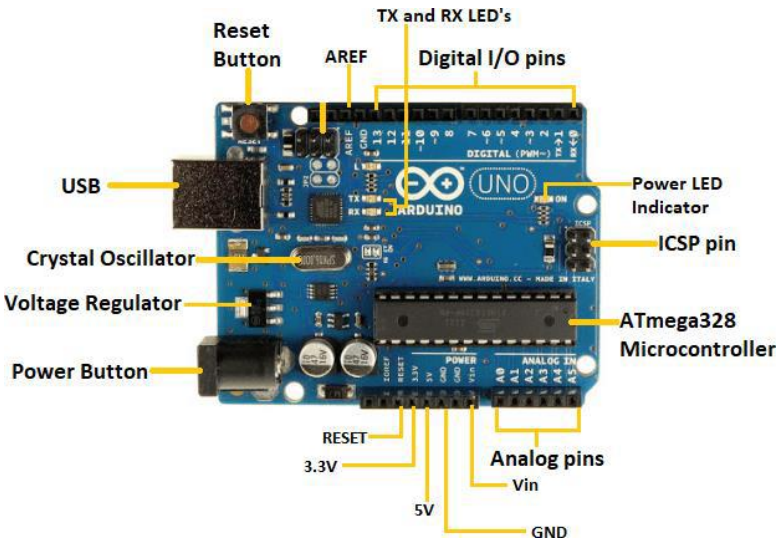


Figure 1: Arduino UNO [1]

3.2. LCD Display:
An LCD display offers live updates to users, presenting real-time information regarding parking space availability.

3.3. Servo Motor:
A servo motor is utilized to control the barrier gate, managing access to the parking lot by either permitting or restricting entry [2].

3.4. Push Button:
A push button switch operates through manual pressure to engage an internal switching mechanism, regulating an electrical circuit. These switches come in various shapes, sizes, and configurations, catering to specific design requirements.

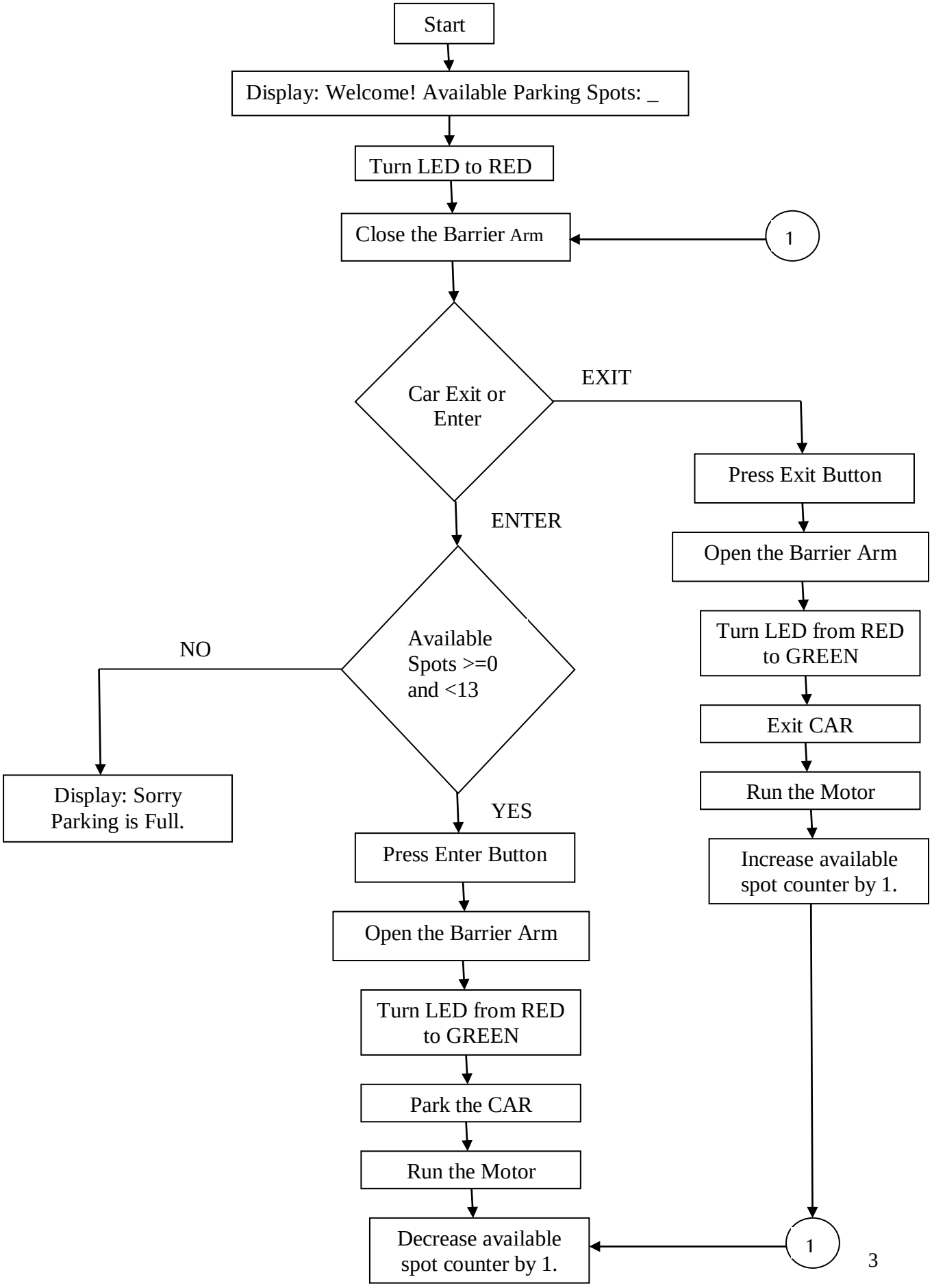
3.5. Battery Module and Connection Wires:
The system's components are powered and interconnected through a designated power source and wiring arrangement.

3.6. RGB LED module:
LED stands for light-emitting diode. These tiny light sources are activated when an electrical current passes through a microchip, emitting visible light. In the project, LEDs are used to indicate the barrier's opening and closing, serve as status indicators for parking spaces, and signal access to the parking lot.

3.7. Breadboard:
A breadboard is a solderless tool employed for constructing temporary electrical prototypes and testing circuit designs. It allows various electrical components in electronic circuits to be linked by inserting their leads or terminals into appropriate holes and, if needed, establishing connections using wires [3].

3.8. Software- MATLAB:
Utilized to develop the project's code, ensuring the system operates in accordance with the specified criteria.

4. Program Logic Design:



5. Circuit Diagram, Design and Function Explanation:

5.1. Circuit Diagram:

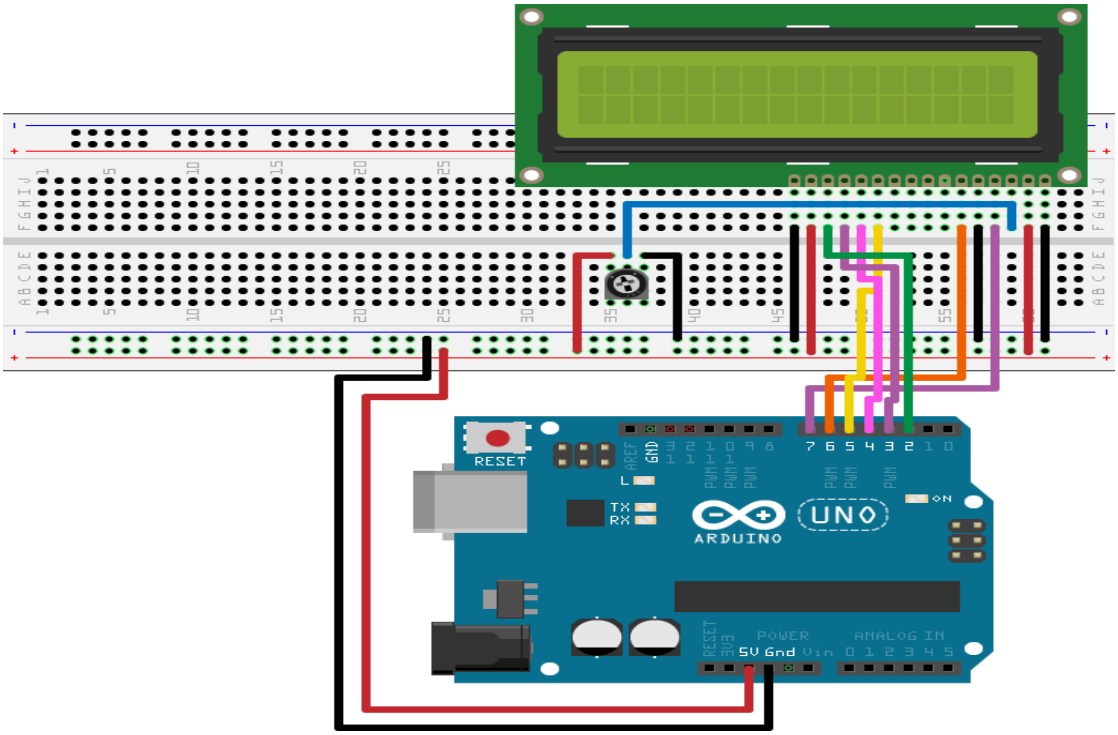


Figure 2: LCD circuit diagram [4]

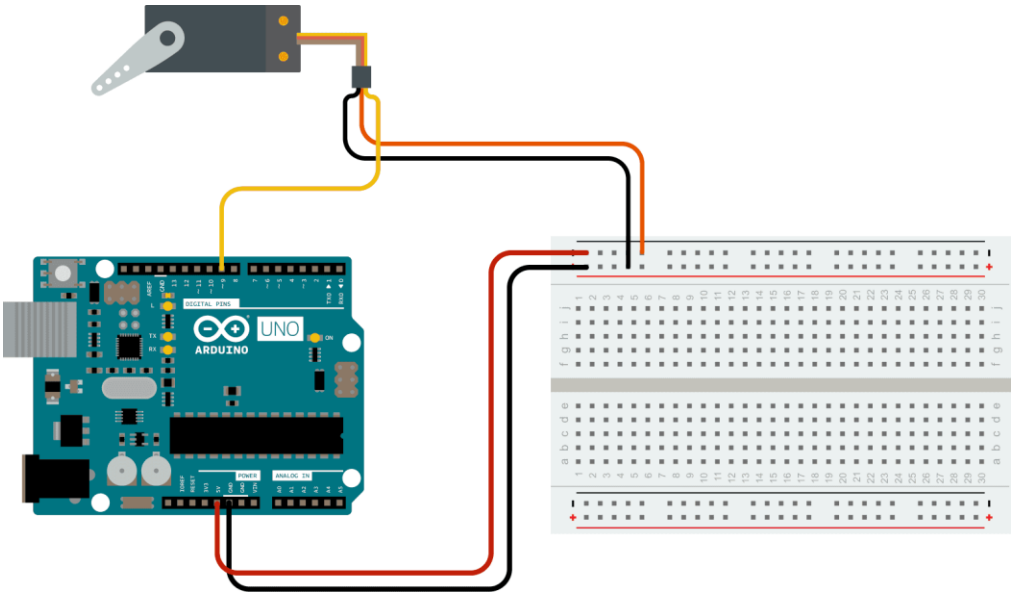


Figure 3: Servo motor circuit diagram [5]

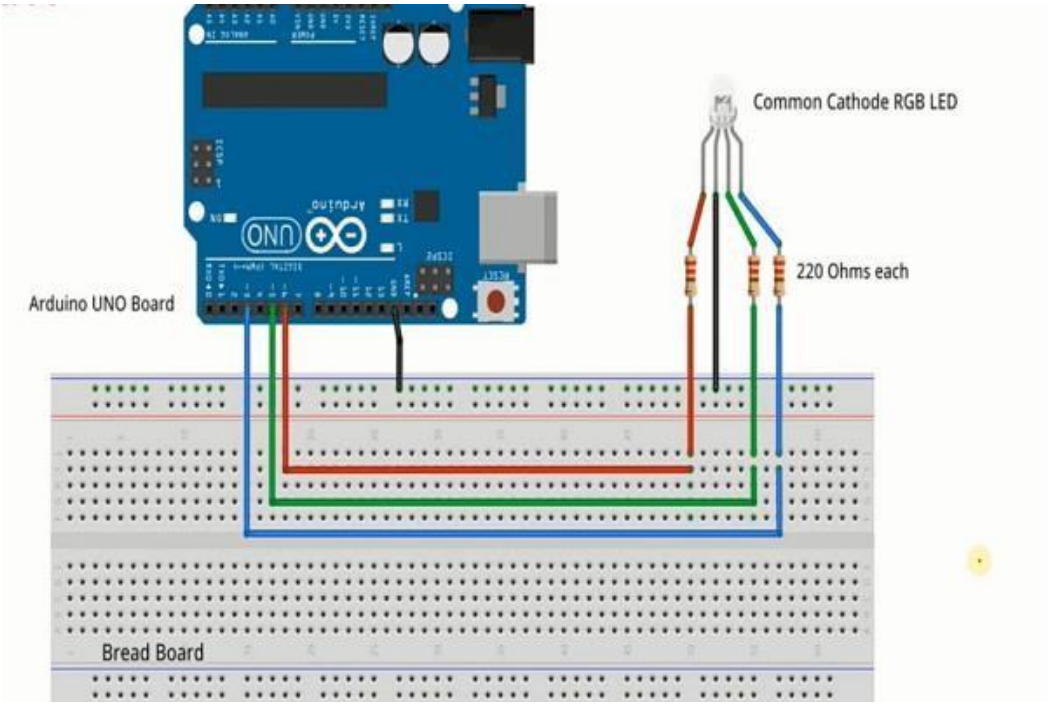


Figure 4: RGB LED circuit diagram [6]

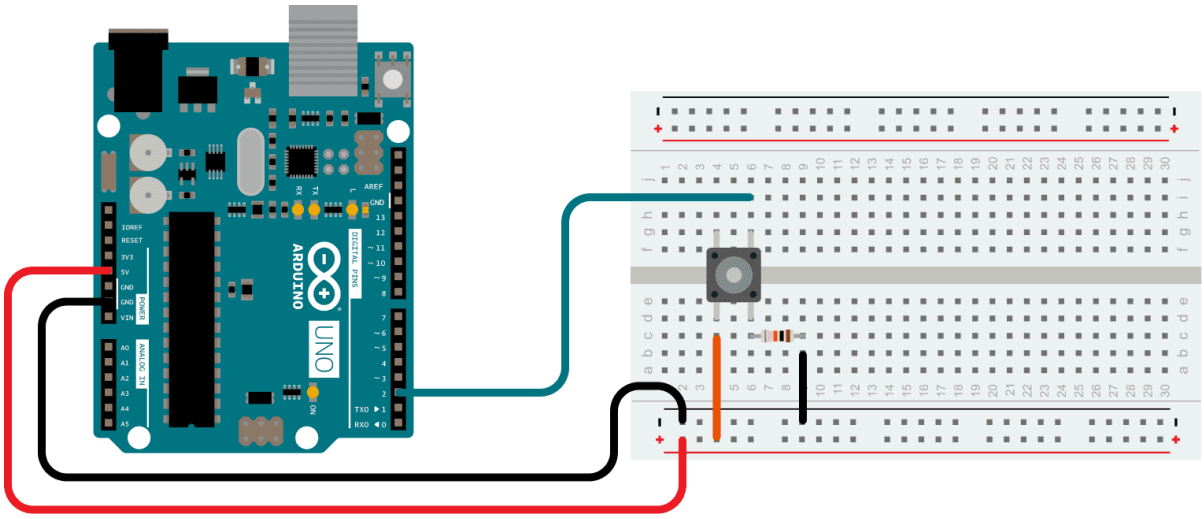


Figure 5: Push button circuit diagram [7]

5.2. Circuit Design:

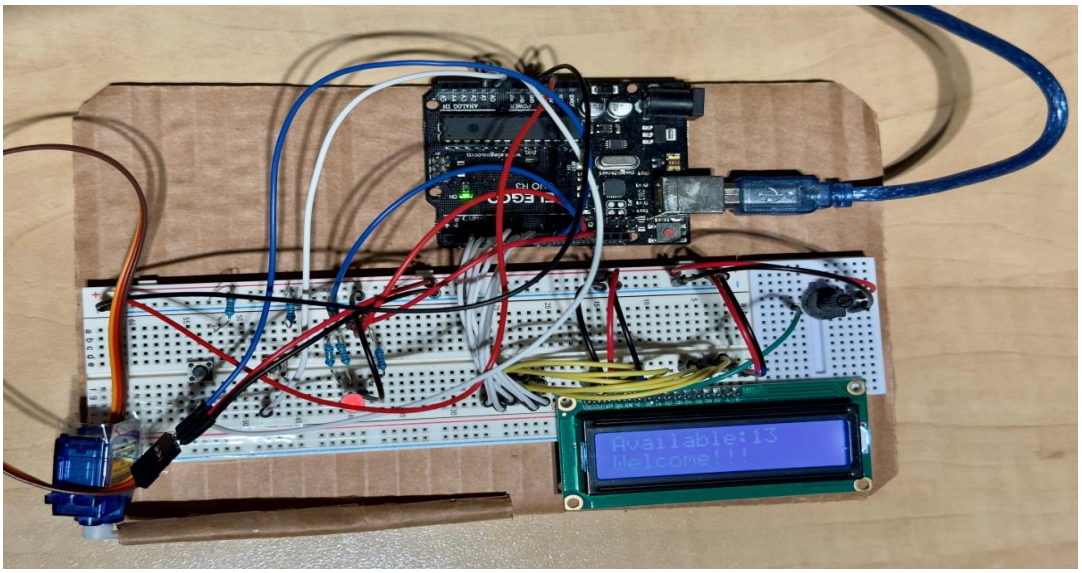


Figure 6: Overall Circuit Design

Figure 5 shows Four basic elements are part of a complete hardware configuration for the project: RGB light emitting diode, LCD 1602 module, push buttons and servo motors.

- There are four pins on the LED: one pin is grounded, and the pin that emits the red light is connected with a resistor to pin D12. At the same time, pin D13 is connected to the pin that produces the green light by a resistor.
- The ENTER and EXIT buttons, which are inputs to allow a driver into or out of the parking space, are represented by two push buttons. These buttons are set up with the negative pin on the other end connected to ground via a resistor and the positive pin on one end connected to 5V power. The positive terminals of the ENTER and EXIT buttons are connected by an arduino board pin A0 and A1.
- The LCD module has 16 pins; pins 1 and 2 are connected to the LCD power supply, while pins 15 and 16 are connected to the backlight. This results in pins 1 and 16 connecting to the ground, followed by Pins 2 and 15 with a 5V connection. The contrast potentiometer is located on pin 3 of the Arduino board, while pin 4 (the register select pin) is connected to pin 7 of the board. In write mode, pin 5 of the LCD, which is used for data read and write, is connected to the ground. At the same time, pin 6 of the LCD is connected to pin 6 of the Arduino, which is the data enabling pin. The data lines D4, D5, D6, and D7 on the LCD are connected to Pins 5, 4, 3, and 2 in the Arduino so that four megabytes of information can be transmitted into the LCD module.
- The servo motor is fitted with three wires: an orange wire connected to the Arduino board, a red and brown connector for the positive and negative terminals. As a result, orange wire will connect to pin 10 of an arduino board, red wire is connected to 5V and brown wires are connected to the ground. In order to enhance the function of the Gate Barrier Arm, a plastic arm shall be added to the motor. The range of motion of the servo motor is the maximum angle of 90 degrees. This limitation ensures precise control by providing a gate barrier arm with the ability to move upwards and downwards as necessary, which will give an accurate picture of the project.

5.3. Function Explanation:

There are, in fact, 13 places available for this specific project. However, it is easy for the user to modify in order to satisfy their own needs because of its flexibility.

The key characteristics of a smart parking system are as follows:
1. The driver can drive to and from the parking lot by means of entry or exit buttons if spaces are available [8]. The word "welcome" appears on the entry. But if you take every slot, the driver is reminded to return later by a message on his LCD screen that says: "Plz come Later."
2. At 0 degrees the servo motor is positioned, signals a closed gate and red LED lights shall be lit first. The servo motor will rotate 90 degrees and the red light will turn green, indicating that the gate is open, when the driver presses the enter button when the slots are open. The servo motor goes back to 0 degrees and the LED lights red as soon as the car enters the parking area, closing the gate. The number of open slots shall be reduced by one at the same time that the LCD panel is updated in relation to new slot details.
3. The driver presses the exit button to start a servo motor that's spinning from 0 to 90 degrees as soon as he pulls out of the parking lot. At the same time, a red LED begins to turn green. When the vehicle leaves the parking lot, the system returns to its original state and adds one to the number of available parking spaces. These changes will be automatically updated and reflected in the LCD panel.

6. Roles and Responsibilities:

Each team member shall be responsible for the project's development and will participate actively in all aspects of its operation, which include completing a final report, developing programming logic or creating MATLAB code to run on an Arduino circuit. Tasks have been allocated on an equal basis depending upon the interest and availability of each team member. Minal had been responsible for assembling the Arduino circuit, as well as all of its components. On the other hand, Nawazhusain's attention was focused on writing a MATLAB script for his final project. At the end, Utkarsh was entrusted with the task of drawing up and compiling the final report.

7. Project Progress and Timeline:

Date	Estimated Project Timeline	Status
02 October	Working with team members to prepare the preliminary report.	Completed
13 October	Discussion of the requirements for the components of the project	Completed
25 October	Designing circuits based on individual components	Completed
07 November	Programming on MATLAB	Completed
14 November	Testing of code and circuits	Completed
21 November	Preparation of the final report and demonstration of the project prior to presentation	Completed

8. References:

[1] “Arduino Uno R3”, [Online] Available: https://www.pololu.com/product/2191 (Accessed: 23-March-2022)

[2] “ MicroServo”, [Online].Available: https://www.pololu.com/product/2817 (Accessed: 23-March-2022)

[3] “Breadboard”, [Online]. Available: https://learn.sparkfun.com/tutorials/how-to-use-a-breadboard/all (Accessed: 23-March-2022)

[4] “Create LCD add-on - MATLAB & Simulink,” *Mathworks.com*, [Online].Available: https://www.mathworks.com/help/supportpkg/arduinoio/ug/add-lcd-library.html.[Accessed 01 December 2022].

[5] “Servo Motor Basics with Arduino,” *Arduino.cc*, [Online]. Available:https://docs.arduino.cc/learn/electronics/servo-motors. [Accessed 01 December2022].

[6] Q.Bruce, “How to Produce Colors With RGB LEDs using Arduino and Matlab Graphical User Interface,” *Engineering Education (EngEd) Program*, [Online]. Available: https://www.section.io/engineering-education/how-to-produce-colors-with-rgb-leds.[Accessed 02 December2022].

[7] “How to wire and program a button,” *Arduino.cc*, [Online]. Available: https://docs.arduino.cc/built-in examples/digital/Button.[Accessed02December2022].

[8] J. Doe, "Smart Parking Solutions: A Review of Current Technologies," IEEE Transactions on Intelligent Transportation Systems, vol. 10, no. 3, pp. 123-135, 2018.