CONTENTS

ABSTRACT

The Smart Parking System (SPS) is an advanced technological solution designed to address the challenges associated with traditional parking management. It leverages various technologies such as sensors, real-time data processing, mobile applications, and connectivity to optimize the parking experience for users. The main objective of this project is to find an empty parking slots in parking area. The primary goal of SPS is to enhance efficiency, reduce traffic congestion, and improve the overall convenience of parking in urban and crowded areas.

Keywords: IR Sensor, Arduino Uno, Servo motor, LCD Display, Smart Parking.

Chapter 1

Introduction

The Internet of Things (IoT) is a revolutionary concept that refers to the interconnectedness of everyday devices and objects through the internet. In IoT, physical objects are embedded with sensors, actuators, and other technologies that enable them to collect and exchange data with other devices and systems over the internet. This interconnected network allows for seamless communication and data sharing between devices, leading to smarter and more efficient processes.

A Smart Parking System is an intelligent and innovative solution designed to address the challenges associated with parking in urban and crowded areas. The traditional approach to parking management often leads to inefficiencies wasted time, and increased traffic congestion. Smart Parking Systems leverage technology to streamline the parking process, optimize space utilization, and enhance the overall parking experience.

The "Smart Parking System" project presented in this report is a creative application of modern technology, combining an Arduino Uno microcontroller, a servo motor, and sensors to develop an intelligent parking system. This report discussed the components that are used for the Smart Parking and the implementation of Smart Parking System using IoT.

Chapter 2

Literature Review

2.1 Internet of Things:

(Internet of Things Applications: Opportunities and Threats by Amir Masoud Rahmani, Suleyman Bayramov & Behnam Kiani Kalejahi Wireless Personal Communications)

In the century of automation, which is digitized, and more and more technology is used, automatic systems' replacement of old manual systems makes people's lives easier. Nowadays, people have made the Internet an integral part of humans' daily lives unless they are insecure. The Internet of Things (IoT) secures a platform that authorizes devices and sensors to be remotely detected, connected, and controlled over the Internet. Due to the developments in sensor technologies, the production of tiny and low-cost sensors has increased. The large number of devices with smart sensors constantly monitor and collect information from the environment, and this data is sent to cloud servers for storage and evaluation. When the user requests information through applications, the processed data is provided to the user quickly and structured. As a result of the development of these sensors with new generations, the power of the IoT technology increases, and accordingly, the revolution of IoT applications are developing rapidly.

2.2 IoT Based Smart Parking System

(Amara Aditya, Shahina Anwarul, Rohit Tanwar, Sri Krishna Vamsi Koneru, Proceedia Computer Science)

The transformation of existing infrastructure into smart cities cannot ignore the smart management of the parking systems. The cities with high population density (Metro cities inclusive) specifically face the problem of finding the nearest available parking space. The rising number of vehicles daily makes this problem more severe that questions the safety of vehicles as well. In this study, the issue of finding parking spaces in smart cities is addressed using an IoT-based methodology. The proposed Intelligent Parking System (IPS) consists of an IoT framework that collects real-time data, send it to the cloud, and thereby suggests to the user a suitable place for parking the vehicle at a nearby location.

In this project, the parking area is constructed with various slots. The IR sensors are equipped with the slots to detect the parking slot availability and the IR sensors are also embedded in the entry and exit of the parking area to detect the vehicle entry and exit in the parking area.

The servo motor is placed at the entry and exit gate that is used to open and close the gates. When a vehicle arrives at the gate of the parking area, the application shows the number of empty slots.

If there is any empty slot in the parking area, the system opens the entry gate by the servo motor. After entering the car into the parking area, it occupies a slot, then it is updated in the mobile app by indicating that the slot is filled.

If there is no empty parking slot, the gate will not be opened by the servo motor. An LCD display is mounted in the entrance of the parking area and it is used to display the availability of parking slots as per the current status of the parking slots.

Chapter 3

Hardware Components

The following hardware elements are required to implement a Smart Parking System. The first section consists of a Arduino Uno, IR sensor, A and a servomotor, all of which are connected to an Arduino using jumper wires. As a result, whenever a car pulls into a parking space, an IR sensor recognizes the presence of the vehicle, sets its output to high, and communicates the information to an Arduino board. In the next section, there is an LCD display that notifies the users about the availability of empty parking slots.

- Arduino Uno
- Infrared Sensor
- Servo Motor
- Bread board
- Jumper Wires

Arduino Uno

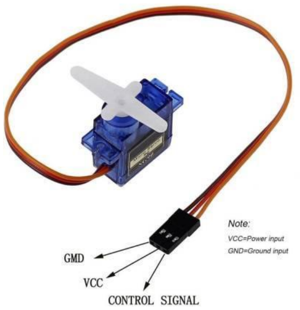
The Arduino Uno is a widespread open-source microcontroller board. It features a variety of digital and analog input/output pins for connecting to various circuits and expansion boards and is based around the Microchip ATmega328P microprocessor. The board has 6 analog I/O pins and 14 digital I/O pins, six of which can generate PWM output. It may be programmed using a Type B USB connector and the Arduino IDE. Within the voltage range of 7 to 20 volts, power can be provided either by a USB cable or an external 9-volt battery.

INFRARED SENSORS(IR)

An Infrared (IR) Sensor is an electronic device that measures and detects infrared radiation in its surrounding environment. Infrared radiation was accidentally discovered by an astronomer named William Herchel in 1800. While measuring the temperature of each color of light (separated by a prism), he noticed that the temperature just beyond the red light was highest.

In this project, one IR detect sensor is used to sense the vehicle near the parking sensor and other IR sensors are used to send data to the Arduino which is the brain of our system whether a vehicle is parked in that slot or not.

SERVO MOTOR



A servomotor is a rotary actuator or linear actuator that allows for precise control of angular or linear position, velocity, and acceleration. It consists of a suitable motor coupled to a sensor for position feedback. The motor is paired with some type of position encoder to provide position and speed feedback. In the simplest case, only the position is measured.

The measured position of the output is compared to the command position, the external input to the controller. If the output position differs from that required, an error signal is generated which then causes the motor to rotate in either direction, as needed to bring the output shaft to the appropriate position. The very simplest servomotors use position-only sensing via a potentiometer and bang-bang control of their motor; the motor always rotates at full speed (or is stopped).

BREADBOARD

A breadboard, solderless breadboard, or protoboard is a construction base used to build semi-permanent prototypes of electronic circuits. Unlike a perfboard or stripboard, breadboards do not require soldering or destruction of tracks. For this reason, breadboards are also popular with students and in technological education.

Compared to more permanent circuit connection methods, modern breadboards have high parasitic capacitance, relatively high resistance, and less reliable connections, which are subject to jostle and physical degradation. Signaling is limited to about 10 MHz, and not everything works properly even well below that frequency.

JUMPER WIRES

A jumper wire (also known as DuPont wire) is an electrical wire, or group of them in a cable, with a connector or pin at each end (or sometimes without them – simply "tinned"), which is normally used to interconnect the components of a breadboard or other prototype or test circuit, internally or with other equipment or components, without soldering.

TYPES OF JUMPER WIRES

Male to Male

This type of male to male Dupont/Jumper Wire 40 Pin 40cm. A very Flexible and easily detachable cable to the no. of wires according to your requirement. It has 1Pin male to the 1pin male header with both ends. Also, it is compatible with 2.54 mm mil spacing pin headers.

Male to Female

These are male to female jumper wires used in connecting the female header pin of any development board to other development boards having a male connector. They are simple wires that have connector pins at each end allowing them to be used to connect two points to each other. Jumper wires are typically used with breadboards and other prototyping tools in order to make it easy to change a circuit as needed.

Female to Female

These are female to female jumper wires used in connecting the female header pin of Arduino or any development board to plug in any other development board. It is reusable. It is inexpensive and easy to use. It is used to interconnect the components of a breadboard or other prototype or test circuit internally or with other equipment or components without soldering.

Chapter 4

Software Setup

4.1. Arduino IDE setup:

1. Download Arduino IDE:

- Visit the official Arduino website at https://www.arduino.cc/en/software.
- Click on the "Download the Arduino IDE" button for Windows.
- Download the installer to your computer.

2. Run the Installer:

- Locate the downloaded installer file (e.g., "arduino-x.x.x-windows.exe").
- Double-click the installer to run it.
- If prompted for administrative permissions, grant them.

3. Installation Wizard:

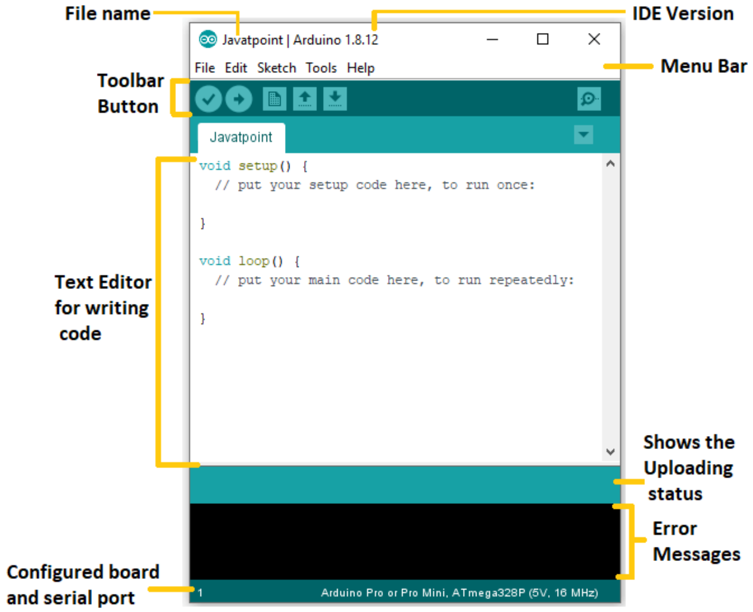
- Follow the on-screen instructions in the installation wizard.
- You can leave most settings at their default values.

4. Driver Installation (if needed):

- During the installation, you may be prompted to install drivers for Arduino boards. Follow the prompts to complete this step.

5. <u>Complete Installation:</u>

- Once the installation is complete, click the "Close" button.


File name
IDE Version
Javatpoint | Arduino 1.8.12
File Edit Sketch Tools Help
Menu Bar
Toolbar Button
Javatpoint
Text Editor for writing code
Shows the Uploading status
Error Messages
Configured board and serial port
1
Arduino Pro or Pro Mini, ATmega328P (5V, 16 MHz)

```
void setup() {
  // put your setup code here, to run once:

}

void loop() {
  // put your main code here, to run repeatedly:

}
```

Arduino IDE

The Arduino IDE is an open-source software, which is used to write and upload code to the Arduino boards. The IDE application is suitable for different operating systems such as Windows, Mac OS X, and Linux. It supports the programming languages C and C++. Here, IDE stands for Integrated Development Environment.

The program or code written in the Arduino IDE is often called as sketching. We need to connect the Genuino and Arduino board with the IDE to upload the sketch written in the Arduino IDE software. The sketch is saved with the extension '.ino.'

Chapter 5

Methodology

5.1. Circuit Diagram:

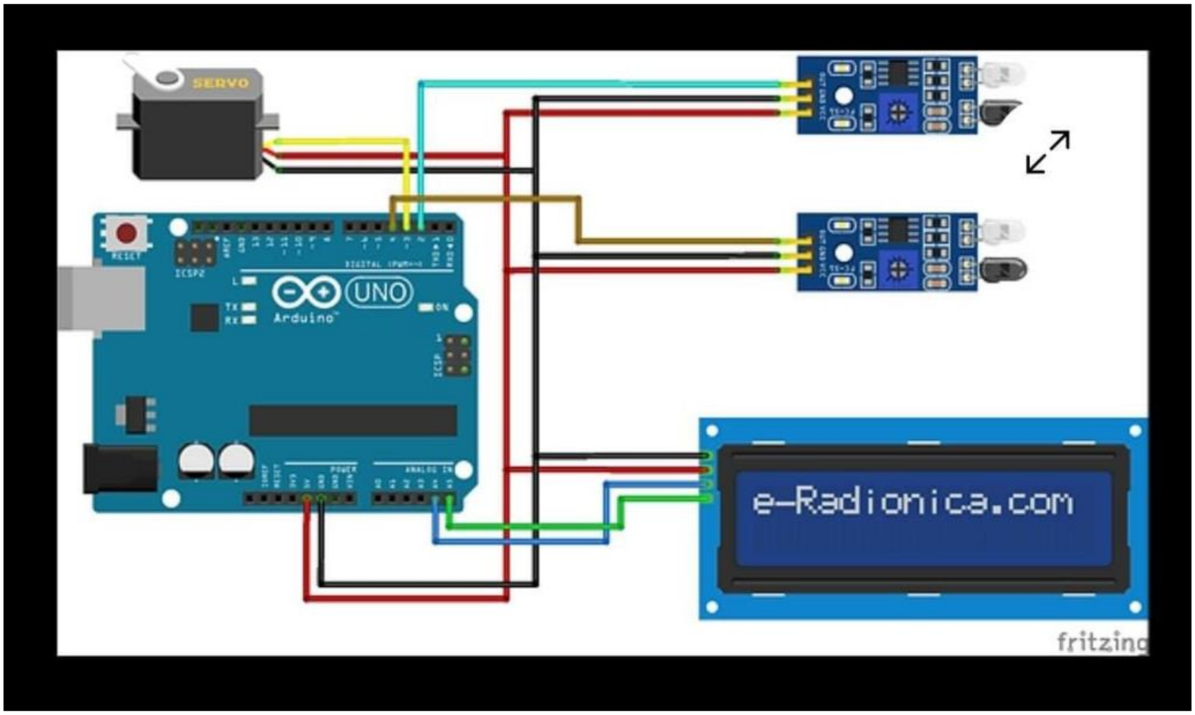


Fig:5.1 Circuit Diagram

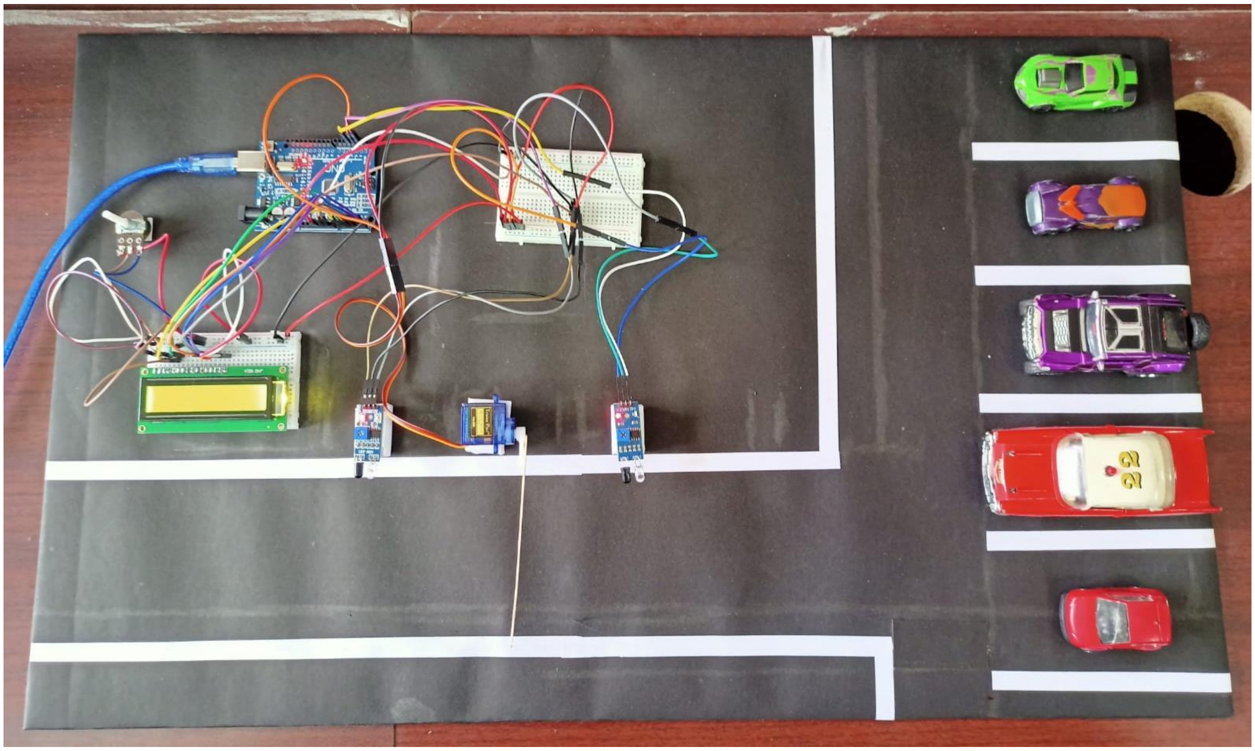
5.2. Configuration with components using code:

```
// Arduino Car Parking System

#include <Wire.h>

#include <LiquidCrystal_I2C.h>

LiquidCrystal_I2C lcd(0x27,16,2);

#include <Servo.h>
```

```
Servo myservo1;
int IR1 = 2;
int IR2 = 4;
int Slot = 4;           //Enter Total number of parking Slots
int flag1 = 0;
int flag2 = 0;

void setup() {
  lcd.init();
  lcd.backlight();
pinMode(IR1, INPUT);
pinMode(IR2, INPUT);
myservo1.attach(3);
myservo1.write(100);

lcd.setCursor (0,0);
lcd.print("     ARDUINO    ");
lcd.setCursor (0,1);
lcd.print(" PARKING SYSTEM ");
delay (2000);
lcd.clear();
}
```

```
void loop(){
if(digitalRead (IR1) == LOW && flag1==0){
if(Slot>0){flag1=1;
if(flag2==0){myservo1.write(0); Slot = Slot-1;}
}else{
lcd.setCursor (0,0);
lcd.print("    SORRY :(    ");
lcd.setCursor (0,1);
lcd.print("  Parking Full  ");
delay (3000);
lcd.clear();
}
}
if(digitalRead (IR2) == LOW && flag2==0){flag2=1;
if(flag1==0){myservo1.write(0); Slot = Slot+1;}
}

if(flag1==1 && flag2==1){
delay (1000);
myservo1.write(100);
flag1=0, flag2=0;
}
lcd.setCursor (0,0);

lcd.print("    WELCOME!    ");

lcd.setCursor (0,1);

lcd.print("Slot Left: ");

lcd.print(Slot);

}
```

5.3. Output Model



Chapter 6

Conclusion and Future Work

Conclusion:

Smart parking system has been proposed to avoid traffic congestion, random parking, and obstruction of traffic in the parking area as well as to reduce the searching and waiting time for a parking space. The growth of Internet of Things and Cloud technologies have given rise to new possibilities in terms of smart cities. Smart parking facilities and traffic management systems have always been at the core of constructing smart cities. In this project, we address the issue of parking and present an IoT based Cloud integrated smart parking system. The proposed project provides real time information regarding availability of parking slots in the parking area. Users from remote locations could book a parking slot for them by the use of our mobile application.

Future Scope:

There is a great scope for the modifications of the system in the future. The system can be improved by adding new functionalities.

FUTURE WORK

The development of more advanced sensors will make it easier for drivers to find available parking spots. These sensors could detect when a car leaves a spot and update the parking availability in real-time.

By using machine learning algorithms, smart parking systems could predict when parking lots will be full and suggest alternative parking options to drivers.