SMART PARKING USING IOT

PHASE-5:

Team Leader name and Members:

VIGNESH SEKAR.S (Leader)

[vigneshsekar2003s@gmail.com](mailto:vigneshsekar2003s@gmail.com)

SYED ALTHAF.F

[althafkhankhan19@gmail.com](mailto:althafkhankhan19@gmail.com)

CHRISTY MATHAI

[christymathai008@gmail.com](mailto:christymathai008@gmail.com)

VASANTHARAM.R

[vasantharam321@gmail.com](mailto:vasantharam321@gmail.com)

Project objectives:

Real-time Parking availability:

Develop a system that can monitor and provide real-time information on parking space availability in a specific area

IOT Device setup:

Setup a network of iot devices,such as sensors or cameras,to collect data on parking space occupancy

Platform Development:

Create a software platform or mobile app that can display the real-time parking data and allow users to find and reserve parking spots

IOT device setup:

Sensor Installation:

Deploy iot sensors such as ultrasonic sensors or cameras in each parking space to detect vehicle presence

Communication protocols:

Ensure these devices are capable of connecting to a central server or cloud platform using communication protocols like MQTT or HTTP

Platform development:

Create a cloud-based infrastructure to collect and process data from data from iot devices.Use platforms like AWS,Azure,or Google cloud

Data Storage:

Store the incoming data in a database,enabling historical analysis and real-time access

Data processing:

Implement data processing to handle outliers,missing data,and data integration from multiple sensors.Use Python and libraries like pandas

Code Implementation:

#define trigPin 2 // Ultrasonic sensor trigger pin

#define echoPin 3 // Ultrasonic sensor echo pin

#define parkingLedPin 13 // LED to indicate parking availability

void setup() {

Serial.begin(9600);

pinMode(trigPin, OUTPUT);

pinMode(echoPin, INPUT);

pinMode(parkingLedPin, OUTPUT);

}

void loop() {

long duration, distance;

digitalWrite(trigPin, LOW);

delayMicroseconds(2);

digitalWrite(trigPin, HIGH);

delayMicroseconds(10);

digitalWrite(trigPin, LOW);

duration = pulseIn(echoPin, HIGH);

distance = (duration / 2) / 29.1; // Calculate distance in centimeters

if (distance < 10) {

// Vehicle detected, parking is occupied

digitalWrite(parkingLedPin, LOW); // LED OFF

Serial.println("Parking Occupied");

} else {

// No vehicle detected, parking is available

digitalWrite(parkingLedPin, HIGH); // LED ON

Serial.println("Parking Available");

}

delay(1000); // Check parking status every 1 second

}

Hardware Components:

Ultrasonic sensors (HC-SR04 or similar) for detecting vehicle presence.

Arduino board (e.g., Arduino Uno or Arduino Nano).

LED indicators to show parking availability.

Optional: LCD display or serial monitor for real-time status.

Arduino Code:

cpp

Copy code

#define trigPin 2 // Ultrasonic sensor trigger pin

#define echoPin 3 // Ultrasonic sensor echo pin

#define parkingLedPin 13 // LED to indicate parking availability

void setup() {

Serial.begin(9600);

pinMode(trigPin, OUTPUT);

pinMode(echoPin, INPUT);

pinMode(parkingLedPin, OUTPUT);

}

void loop() {

long duration, distance;

digitalWrite(trigPin, LOW);

delayMicroseconds(2);

digitalWrite(trigPin, HIGH);

delayMicroseconds(10);

digitalWrite(trigPin, LOW);

duration = pulseIn(echoPin, HIGH);

distance = (duration / 2) / 29.1; // Calculate distance in centimeters

if (distance < 10) {

// Vehicle detected, parking is occupied

digitalWrite(parkingLedPin, LOW); // LED OFF

Serial.println("Parking Occupied");

} else {

// No vehicle detected, parking is available

digitalWrite(parkingLedPin, HIGH); // LED ON

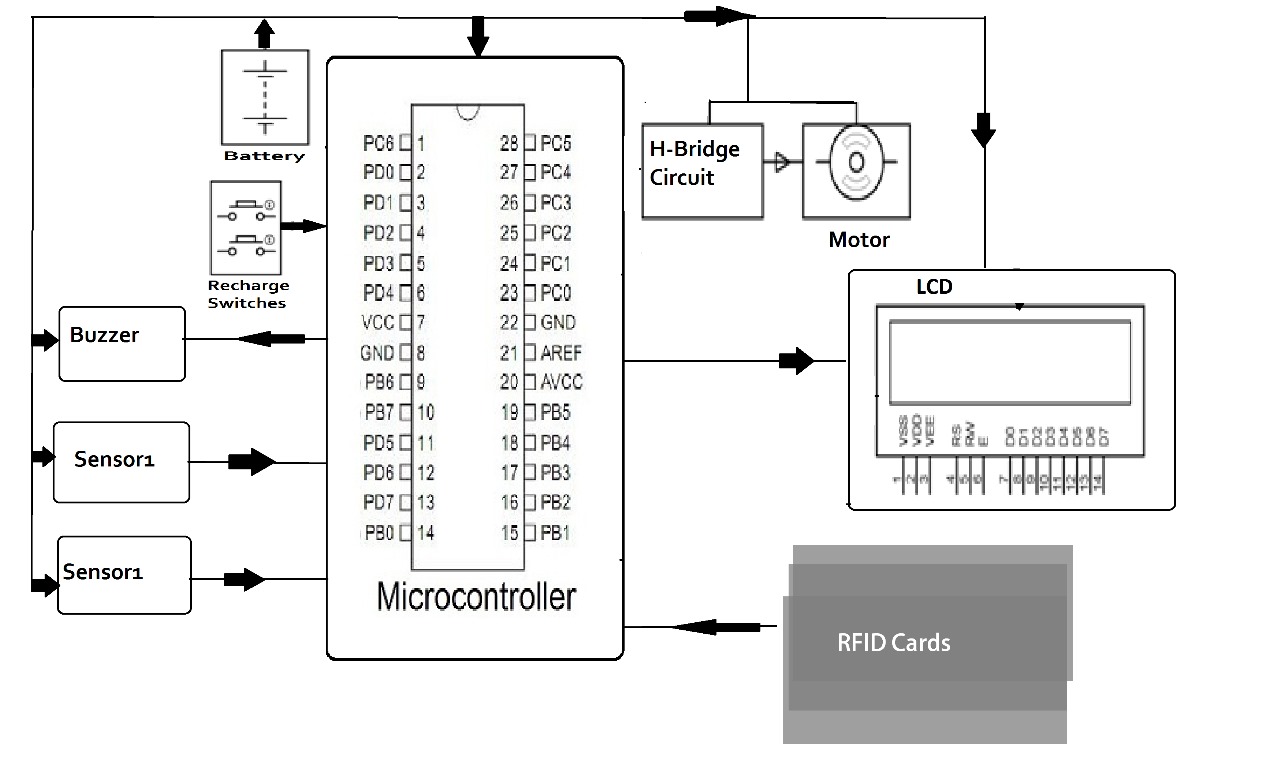
Serial.println("Parking Available");

}

delay(1000); // Check parking status every 1 second

}

Diagram:



Schematics:

Traffic Management Integration:

Integration with city traffic management systems to impove traffic flow and urban planning

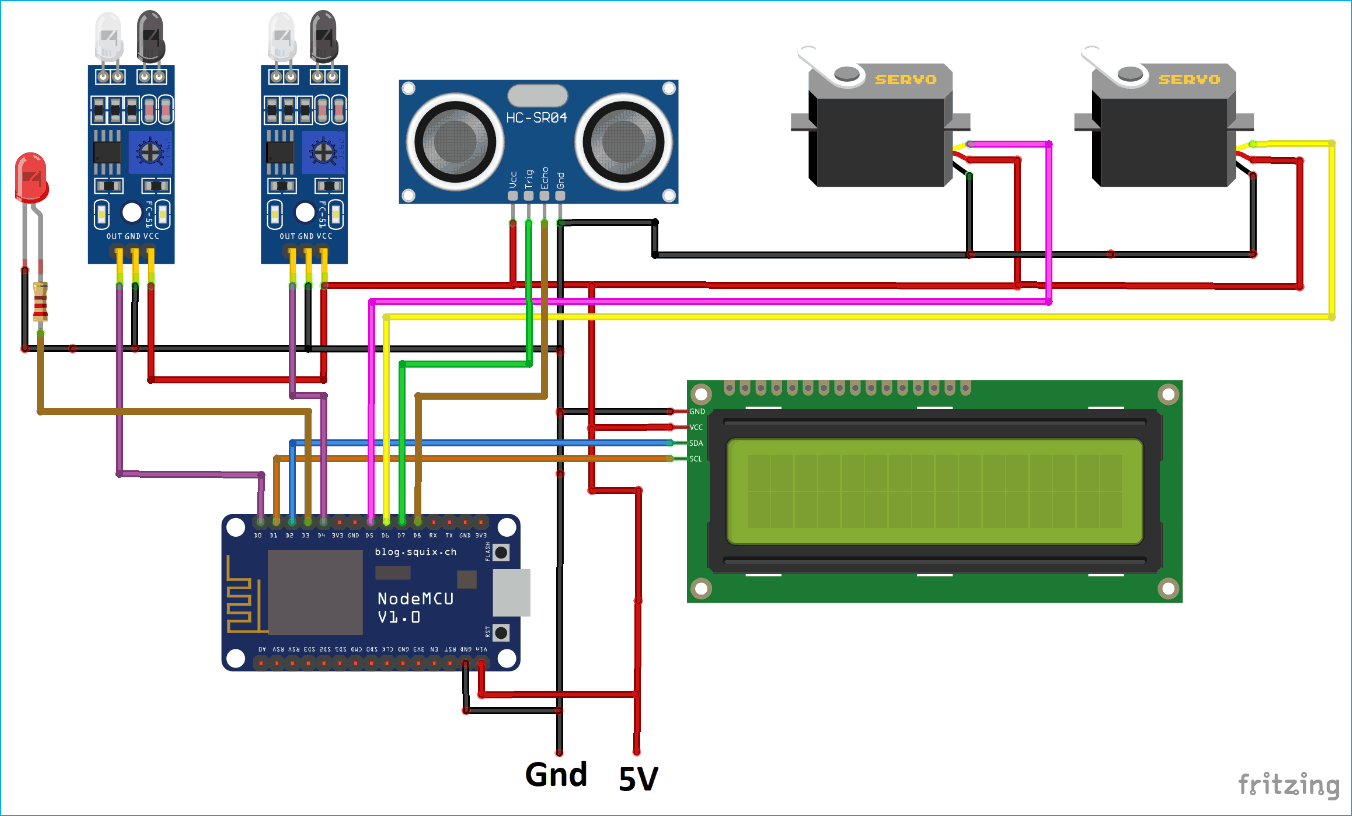
User Access control:

User registration and provide feedback or report issues

Data processing:

Implement data processing to handle outliers,missing data,and data integration from multiple sensors.Use Python and libraries like pandas

Screenshots:



Data Sharing platform Development:

Create user-friendly web or mobile application for users to access parking information.This platform can be developed using programming languages like Python,javascript or any suitable framework

Conclusion:

As iot technology continues to advance,smart parking systems will become increasingly sophisticated and integrated into the broader ecosystem of smart cities