**Smart Parking System using ESP32, IR Sensors, and I2C LCD**

In modern urban environments, finding available parking spaces has become a growing challenge due to increasing vehicle density and lack of real-time monitoring. This often results in traffic congestion, fuel wastage, and driver frustration. To address this issue, we designed a Smart Parking System using an ESP32 microcontroller, IR sensors, and an I2C-based LCD display to monitor and display the real-time status of parking slots.

The core idea of this project is to detect whether a parking slot is free or occupied and display the status to the user. The system utilizes three IR proximity sensors, each assigned to a specific parking slot. One of these sensors, specifically the one assigned to Slot 1, is simulated using a push button for demonstration in the Wokwi simulator. The IR sensors work on a simple principle: when a vehicle is present in the slot, it obstructs the IR beam, causing the sensor to output a LOW signal, which is interpreted as "Occupied." When the slot is empty, the sensor outputs HIGH, which is displayed as "Free."

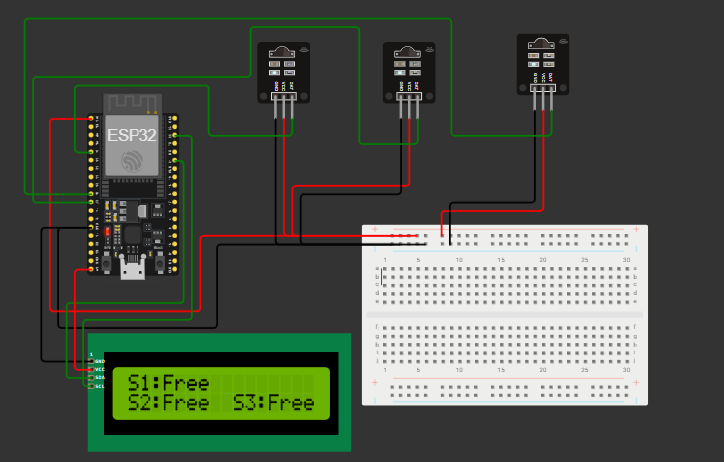
The ESP32 serves as the brain of the system. It continuously reads the status of each sensor and updates the 16x2 I2C LCD display in real-time. The LCD is connected via the I2C protocol using GPIO21 (SDA) and GPIO22 (SCL), which simplifies wiring and conserves ESP32 pins. The slot status is presented on the LCD as: "S1:Free", "S2:Occupied", "S3:Free", depending on the actual sensor readings. This allows users to quickly identify which slots are available.

In the simulation environment (Wokwi), a push button connected to GPIO34 is used to mimic an IR sensor for Slot 1. When the button is not pressed, the input reads HIGH (indicating the slot is free). When the button is pressed, it pulls the input LOW (indicating the slot is occupied). The remaining two sensors are treated as regular IR sensors and are connected to GPIO27 and GPIO26 respectively.

This project demonstrates a practical, low-cost solution to a real-world problem. It can be easily scaled to accommodate more slots by adding additional sensors and modifying the code accordingly. Moreover, this system lays the foundation for future improvements such as web or mobile dashboard integration, cloud-based remote monitoring using platforms like Firebase, or automatic gate control based on slot availability.

In conclusion, the Smart Parking System using ESP32, IR sensors, and an I2C LCD offers an efficient, real-time, and scalable solution to parking space detection. It is ideal for applications in malls, corporate campuses, residential societies, and public parking areas. The project helps reduce congestion and promotes smart city development by making parking more efficient and user-friendly.

CIRCUIT DIAGRAM :



CODE:

#include <Wire.h>

#include <LiquidCrystal\_I2C.h>

LiquidCrystal\_I2C lcd(0x27, 16, 2);

#define SLOT1\_PIN 34

#define SLOT2\_PIN 27

#define SLOT3\_PIN 26

void setup() {

  pinMode(SLOT1\_PIN, INPUT\_PULLUP);

  pinMode(SLOT2\_PIN, INPUT);

  pinMode(SLOT3\_PIN, INPUT);

  lcd.init();

  lcd.backlight();

  lcd.clear();

  lcd.setCursor(0, 0);

  lcd.print("Smart Parking");

  delay(2000);

  lcd.clear();

}

void loop() {

  bool slot1 = digitalRead(SLOT1\_PIN);

  bool slot2 = digitalRead(SLOT2\_PIN);

  bool slot3 = digitalRead(SLOT3\_PIN);

  lcd.setCursor(0, 0);

  lcd.print("S1:");

  lcd.print(slot1 ? "Free    " : "Occupied");

  lcd.setCursor(0, 1);

  lcd.print("S2:");

  lcd.print(slot2 ? "Free  " : "Occ ");

  lcd.print("S3:");

  lcd.print(slot3 ? "Free" : "Occ ");

  delay(500);

}