**Project Title: Smart Parking System Using ATmega32**

**Objective:**

To create a simple automated parking monitoring system that detects the presence of cars in parking slots using IR sensors and indicates the slot status through LEDs using the ATmega32 microcontroller.

**Working Principle:**

The system monitors **four parking slots**. Each slot uses an **IR sensor** to detect whether a car is present. If a car is detected, the corresponding **LED** turns **ON**, indicating that the slot is occupied. If there is no car, the LED remains **OFF**, showing the slot is empty.

The **ATmega32 microcontroller** continuously reads input signals from the IR sensors and controls the LED output accordingly. The logic is handled through simple bit-checking instructions and efficient delay routines that simulate the real-time reaction of the system.

**Components Used:**

* ATmega32 Microcontroller
* 4 IR Sensors (Obstacle/Car detection)
* 4 LEDs (for indicating status)
* Resistors (220 ohms for LEDs)
* Power Supply (5V regulated)
* Proteus (for simulation)
* AVR Studio / MikroC / Atmel Studio (for writing and uploading code)

**Pin Configuration:**

| **Component** | **Connected To** | **Description** |
| --- | --- | --- |
| IR Sensor 1 | **PIND0 (pin 14)** | Input for Slot 1 |
| IR Sensor 2 | **PIND1 (pin 15)** | Input for Slot 2 |
| IR Sensor 3 | **PIND2 (pin 16)** | Input for Slot 3 |
| IR Sensor 4 | **PIND3 (pin 17)** | Input for Slot 4 |
| LED for Slot 1 | **PORTC0 (pin 22)** | Output LED for Slot 1 |
| LED for Slot 2 | **PORTC1 (pin 23)** | Output LED for Slot 2 |
| LED for Slot 3 | **PORTC2 (pin 24)** | Output LED for Slot 3 |
| LED for Slot 4 | **PORTC3 (pin 25)** | Output LED for Slot 4 |

* **PORTD** is configured as **input** (IR sensor signals).
* **PORTC** is configured as **output** (LED control).

**Code Logic Summary:**

1. **Initialization:**
   * All **PORTD** pins set as **input**.
   * Lower 4 bits of **PORTC** set as **output** to control 4 LEDs.
2. **Main Loop:**
   * Continuously read the state of PORTD (sensor input).
   * Check each bit (PIND0–PIND3):
     + If the bit is **clear (0)**, a car is present → **Turn ON** LED.
     + If the bit is **set (1)**, no car → **Turn OFF** LED.
   * Very **short delay (~100ms)** is used when car is detected (entering).
   * Slightly longer **delay (~150ms)** when the car leaves the slot.
3. **Delays:**
   * Implemented using loop counters.
   * Fine-tuned to provide minimal lag in sensor response.

**How It Works in Real-Time:**

* IR sensors output **LOW (0)** when a car is detected, and **HIGH (1)** when no car is present.
* The microcontroller logic uses **SBRC/SBRS** instructions to check each bit in the input register.
* If bit is **clear** (car present), it **sets** the corresponding PORTC pin → LED turns ON.
* If bit is **set** (no car), it **clears** the PORTC pin → LED turns OFF.

**Applications:**

* Small parking lots
* Basement or indoor car parks
* Prototype for smart city parking solutions
* Educational demonstration of embedded systems and sensors