Smart Parking System using Arduino

In a Smart Parking System using Arduino, the main goal is to automate the process of detecting the availability of parking spots and controlling the entry and exit gates of a parking area. The components you've listed—Arduino, two servo motors, four PIR sensors, a potentiometer, a resistor, and an LCD display—each play a unique and interconnected role in making this system functional and efficient.

At the heart of the circuit is the **Arduino**, which acts as the brain of the system. It takes input from the PIR sensors, processes the data, and sends appropriate output signals to control the servo motors and update the LCD display. The **PIR (Passive Infrared) sensors** are used to detect the presence or absence of a vehicle in parking spots. Since you have four PIR sensors, each one can be assigned to a specific parking space. These sensors work by detecting motion or heat from a nearby object, in this case, a car. When a car enters a spot, the corresponding PIR sensor detects its presence and sends a digital HIGH signal to the Arduino. When the car leaves, the signal returns to LOW.

The system also has **two servo motors**, which are typically used to control barriers or gates at the entrance and exit of the parking lot. The Arduino controls these servo motors based on the data received from the sensors. For instance, if there's at least one empty spot, the Arduino will rotate the entrance servo motor to open the gate, allowing a vehicle to enter. After a short delay, the motor returns the gate to the closed position. Similarly, when a car is detected near the exit by another sensor (possibly one of the four, or a dedicated one), the Arduino triggers the second servo motor to open the exit gate.

To help the user interact with the system or monitor its status, an **LCD display** is included. This display is used to show real-time information about the number of available parking spots. For example, if three out of four sensors detect a car, the LCD might display “Available: 1” or something similar. The Arduino continuously checks the status of each PIR sensor and updates the LCD accordingly. This helps drivers quickly determine whether there is space to park before even entering.

The **potentiometer** is typically used to adjust the brightness or contrast of the LCD screen. It is connected to the V0 (contrast) pin of the LCD and allows you to manually tune how visible the text on the screen is. This is useful in different lighting conditions to ensure that the display is always readable. The **resistor**, in this setup, is likely used in series with an LED (maybe the backlight of the LCD or a status indicator) to limit current and prevent damage to the components. It could also be part of a pull-down configuration for a sensor input if needed.

All the components are wired in a way that they continuously work together. The PIR sensors constantly send data to the Arduino about whether parking spaces are occupied. Based on this data, the Arduino updates the LCD and decides whether to allow entry or not by controlling the entrance gate servo. Similarly, when a car wants to exit, the exit gate servo is activated accordingly. The system keeps updating in real time, making it an effective smart solution for managing limited parking space.