**Arduino Self-Driving Car Project**

**1. Overview**

This project demonstrates a **self-driving car** using an **Arduino** microcontroller, **four stepper motors**, and **four ultrasonic distance sensors**. The car can detect and avoid obstacles, make turns, and even back up when needed, showcasing basic autonomous navigation.

**2. Key Hardware Components**

1. **Arduino Board**
   * Acts as the main controller, reading sensor data and sending commands to the motors.
   * Common choices include Arduino UNO, Mega, or Nano.
2. **Stepper Motors (28BYJ-48)**
   * Two stepper motors for differential drive (left and right wheels).
   * Provides precise control over movement and turning angle.
   * Each motor is typically driven by a dedicated ULN2003 driver board.
3. **Ultrasonic Distance Sensors (4x)**
   * Placed on the front, back, left, and right sides of the car.
   * Continuously measure distances to detect obstacles.
   * Data is used to decide turning, braking, or reversing actions.
4. **Power Supply**
   * Ensure enough voltage for Arduino logic (5V) and sufficient current for the motors.
   * If using AA batteries, consider 6–9V input through Vin or a DC-DC regulator to maintain a stable 5V output.

**3. Software & Control Logic**

1. **Main Control Loop**
   * The loop() function reads ultrasonic sensors and makes real-time decisions.
   * Uses Arduino’s built-in functions like millis() for timing and scheduling.
   * All sensor sampling, obstacle detection, and motion commands happen within a single-threaded loop, although carefully managed to avoid blocking.
2. **Distance Measurement**
   * Each ultrasonic sensor is triggered in turn to avoid interference.
   * A **non-blocking or short-timeout** measurement method can be used to reduce motor “stutter.”
   * Sensor readings are filtered to handle noise or invalid (zero) results.
3. **Obstacle Avoidance**
   * If any sensor detects an object closer than the **safe distance** (e.g., 25cm), the Arduino decides how to react:
     + Turn left or right if a side is clear.
     + Reverse if both sides are blocked.
     + Stop completely if 4 directions are blocked.
4. **Motor Control**
   * Each stepper motor is driven by four digital signals from Arduino to the ULN2003 board.
   * By adjusting step pulses and direction pins, the car can move forward, reverse, or pivot.
   * Libraries like AccelStepper can enable smoother motion and speed ramps.
5. **Filtering & Error Handling**
   * Small or zero readings from sensors (often noise or blind-zone measurements) are either ignored, marked as invalid, or capped to a minimum distance.
   * If needed, multiple readings can be averaged or median-filtered to increase stability.