**Ultrasonic Radar with Arduino**

**Abstract:**

This project outlines the development of an ultrasonic radar system using an Arduino Uno, an ultrasonic sensor (HC-SR04), and a servo motor. This radar can detect objects within its range and visualize them using Processing software.

**Components and Materials:**

**Hardware:**

* Arduino Uno
* SG90 Micro-servo motor
* Ultrasonic Sensor - HC-SR04 (Generic)
* Jumper wires
* Hot glue gun (optional)

**Software:**

* Arduino IDE (Integrated Development Environment)
* Processing IDE

**Methodology:**

This ultrasonic radar system operates by:

1. **Mounting the Ultrasonic Sensor:** The HC-SR04 sensor is attached to the servo motor using hot glue (or a more permanent method) to enable scanning in different directions.
2. **Sensor Connection:**

The HC-SR04 connects to the Arduino Uno as follows:

* VCC (sensor) to 5V (Arduino)
* GND (sensor) to GND (Arduino)
* TRIG (sensor) to pin 2 (Arduino) - trigger pin.
* ECHO (sensor) to pin 3 (Arduino) - echo pin.

1. **Servo Motor Connection:**

The servo motor connects to the Arduino Uno as follows:

* Red wire (servo) to 5V (Arduino) – power.
* Brown wire (servo) to GND (Arduino) – ground
* Orange wire (servo) to pin 4 (Arduino) - control pin.

1. **Software Functionality:**

**Arduino Code:**

* Defines a function to measure distance using the ultrasonic sensor's time-of-flight principle.
* **Implements a loop to continuously:**

1. Rotate the servo motor from a start angle to an end angle (scanning range).

2. Measure the distance to any object detected by the sensor in its path.

3. Send this distance data to the Processing software.

**Processing Code:**

1. Receives distance data from the Arduino via the serial port.
2. **Visualizes the radar display based on the received distances:**
3. Green lines represent no object detected (distance beyond a threshold).
4. Red lines represent object detected (distance within the threshold).

**Building the Project:**

1. **Hardware Connection:** Connect the ultrasonic sensor and servo motor to the Arduino according to the pin assignments mentioned earlier.
2. **Coding:**
3. Develop the Arduino code with the distance measurement function and the servo motor control loop.
4. Write the Processing code for receiving data and visualizing the radar display.
5. **Testing and Debugging:**
6. Upload the Arduino code to the Arduino board.
7. Run the Processing code.
8. Test the system by placing objects at different distances from the sensor and observe the changes in the radar display.

**Results and Observations:**

The project successfully builds a functional ultrasonic radar system. The system effectively detects objects within its range and visualizes them in the Processing window. The color-coded lines (green for no object, red for object detected) provide a clear indication of the object's presence and location.

**Future Improvements:**

* **Accuracy and Range:** Implement calibration techniques to improve sensor accuracy and explore extending the detection range.
* **Object Classification:** Utilize additional sensors (e.g., infrared) to differentiate between different types of objects.
* **Advanced Display:** Develop a more sophisticated display using graphical libraries or a dedicated LCD screen.

**Conclusion:**

This project demonstrates the capabilities of Arduino in building a simple yet practical ultrasonic radar system. The project provides a valuable learning experience in interfacing sensors, motors, and software for real-world applications.