Here’s a **detailed documentation** for your **solar tracking system** using an **LDR (Light Dependent Resistor)** and **Arduino Uno**. This document covers the **objective**, **components**, **working principle**, **circuit diagram**, **code**, and **testing process**.

**🌞 Solar Tracking System using LDR and Arduino Uno**

**📌 1. Objective**

The objective of this project is to build a simple, cost-effective **solar tracking system** that continuously aligns a solar panel to the direction of **maximum sunlight** using **LDR sensors** and an **Arduino Uno**. This enhances the efficiency of solar energy capture compared to a static panel.

**🧰 2. Components Required**

| **Component** | **Quantity** |
| --- | --- |
| Arduino Uno | 1 |
| LDR (Light Dependent Resistors) | 2 |
| 10kΩ Resistors (for LDR voltage divider) | 2 |
| Servo Motor (e.g., SG90 or MG996R) | 1 |
| Breadboard | 1 |
| Jumper Wires | As needed |
| Solar Panel (optional) | 1 |
| Cardboard or plastic mount | 1 |
| Power Supply (USB or external 5V) | 1 |

**⚙️ 3. Working Principle**

The system uses two LDRs to detect light intensity from two directions (left and right). These sensors are placed on either side of a small barrier (to block central light) and mounted on a platform with a servo motor that simulates the solar panel.

* If the **left LDR detects more light**, the system rotates the servo to the left.
* If the **right LDR detects more light**, the system rotates the servo to the right.
* If the light levels are approximately equal, the servo motor holds position.

This simulates the behavior of a solar panel that always faces the sun directly.

**🔌 4. Circuit Diagram**

**LDR Circuit (Voltage Divider Setup)**

Each LDR is connected in a voltage divider configuration:

* Connect one end of each LDR to **5V**.
* Connect the other end to **analog pins A0 and A1** respectively.
* Between the LDR and **GND**, connect a **10kΩ resistor**.
* This allows you to read voltage changes on the analog pins based on light intensity.

**Servo Motor Setup**

* **Signal Pin** to **D9** (PWM pin)
* **VCC** to **5V**
* **GND** to **GND**

**Pin Summary:**

| **Component** | **Arduino Pin** |
| --- | --- |
| LDR1 | A0 |
| LDR2 | A1 |
| Servo | D9 |

**🧾 5. Arduino Code**

cpp

CopyEdit

#include <Servo.h>

Servo solarServo;

const int ldrLeft = A0;

const int ldrRight = A1;

int pos = 90; // Initial servo position

void setup() {

solarServo.attach(9); // Attach servo to pin 9

solarServo.write(pos);

Serial.begin(9600);

}

void loop() {

int leftValue = analogRead(ldrLeft);

int rightValue = analogRead(ldrRight);

int threshold = 50; // Sensitivity adjustment

Serial.print("Left: "); Serial.print(leftValue);

Serial.print(" | Right: "); Serial.println(rightValue);

if (abs(leftValue - rightValue) <= threshold) {

// Light is balanced

return;

}

if (leftValue > rightValue) {

pos += 1;

if (pos > 180) pos = 180;

} else {

pos -= 1;

if (pos < 0) pos = 0;

}

solarServo.write(pos);

delay(100);

}

**🛠️ 6. Assembly and Mounting**

1. **Mount the LDRs** on either side of a small vertical barrier to ensure light from one side does not affect the other.
2. **Attach the servo motor** securely to the base of your system.
3. **Mount a small cardboard or solar panel** platform on the servo horn to simulate panel movement.
4. **Ensure wiring is secure** and won't disconnect during movement.

**🧪 7. Testing and Calibration**

* **Test indoors and outdoors** under different light conditions.
* Adjust the threshold value in code to suit your LDR sensitivity.
* Ensure that the servo doesn't jitter by fine-tuning logic to reduce unnecessary movement.

**📝 8. Troubleshooting Tips**

| **Problem** | **Solution** |
| --- | --- |
| Servo doesn’t move | Check power connections and servo pin |
| LDR readings not changing | Make sure LDRs are not shorted, test with a flashlight |
| Movement is jerky or too fast | Add delay(100) or increase threshold |
| Servo moves in wrong direction | Reverse the logic in the if conditions |

**🌐 9. Extensions and Improvements**

* Add a second axis for **dual-axis tracking** (horizontal + vertical).
* Use 4 LDRs for more precise alignment.
* Log data to an SD card or transmit via Bluetooth/Wi-Fi.
* Power the system using the actual solar panel output.

**📷 10. Visualization (Optional)**

If you need, I can help create:

* A **circuit diagram**
* A **3D model sketch**
* Or visuals for your project report or presentation

Let me know if you'd like that!