SAVEETHA SCHOOL OF ENGINEERING
SAVEETHA INSTITUTE OF MEDICAL AND TECHNICAL SCIENCES

MINI PROJECT REPORT

ECA1420 AND Embedded Systems for Smart Applications

<u>PROJECT TITLE</u>

Solar Energy Maximization Tracker

Submitted
by
Y. YAMINI REDDY (192211153)
G. SHREENIDHI (192211161)

Guided by

Dr. Sourav Ghosh
Assistant Professor
Embedded System

ABSTRACT

The solar sun tracking system is a cutting-edge technology designed to enhance the efficiency of solar energy collection by dynamically aligning solar panels with the sun's position throughout the day. Unlike conventional fixed solar panel systems, which capture limited energy due to the sun's changing position in the sky, sun tracking systems significantly increase energy output by ensuring that panels are continuously oriented at an optimal angle to receive maximum sunlight. This approach directly addresses one of the key challenges in solar energy utilization—maximizing efficiency without increasing the size of the installation.

The system integrates various components, including light sensors, microcontrollers, and mechanical actuators, to enable precise and continuous adjustments to the orientation of the solar panels. Sensors, such as light-dependent resistors (LDRs), or algorithms based on astronomical calculations, determine the sun's position. This data is processed by a microcontroller, which then controls motors to adjust the panel's tilt and azimuth angles. Dual-axis tracking systems, which account for both daily east-to-west movement and seasonal changes in the sun's altitude, provide even greater efficiency than single-axis trackers.

The development of this system involves addressing key engineering challenges, including energy consumption of the tracking mechanism, environmental durability, and cost-effectiveness. By incorporating efficient designs and leveraging advancements in materials and control systems, modern sun tracking systems are increasingly accessible and reliable.

The implementation of solar sun tracking systems has profound implications for the renewable energy sector. By maximizing solar panel efficiency, these systems reduce the levelized cost of energy (LCOE) and accelerate the adoption of solar power as a viable alternative to fossil fuels. Their widespread use contributes to sustainable development goals by minimizing carbon emissions and supporting energy independence. This expanded abstract explores the operational principles, design considerations, and broader impact of solar sun tracking systems, highlighting their potential to transform the renewable energy landscape.

INTRODUCTION

Solar energy is one of the most abundant and clean sources of renewable energy, offering a sustainable solution to meet the world's growing energy demands. However, the efficiency of solar power systems depends significantly on how effectively solar panels capture sunlight. Fixed solar panels, though widely used, are limited by their inability to adapt to the sun’s movement throughout the day and across seasons. This limitation results in suboptimal energy production, particularly during mornings, evenings, and winter months when sunlight falls at steeper angles.

To overcome this challenge, solar tracking systems have been developed to dynamically adjust the orientation of solar panels to follow the sun’s path. These systems ensure that solar panels maintain an optimal angle relative to the sun, thereby maximizing energy absorption throughout the day. By employing mechanisms that track the sun's position, either through sensors or computational algorithms, solar tracking systems can significantly enhance the energy yield of photovoltaic systems and solar thermal collectors.

Solar tracking technology comes in two main configurations: single-axis and dual-axis tracking systems. Single-axis trackers move panels along one direction (east to west), while dual-axis trackers adjust both tilt and rotation to follow the sun more precisely. While these systems involve higher initial costs and additional mechanical components, the increased energy output often offsets these drawbacks, making them an attractive option for both large-scale solar farms and smaller installations.

This introduction explores the fundamental concepts, types, and advantages of solar tracking systems, setting the stage for a deeper understanding of their design, operation, and potential contributions to renewable energy solutions. By addressing the limitations of fixed solar installations, sun tracking systems play a crucial role in advancing solar power efficiency and supporting global efforts toward sustainable energy development.

LITERATURE REVIEW

The development of solar sun tracking systems has been a subject of extensive research, as they offer a promising solution to enhance the efficiency of solar energy systems. This literature review examines key studies and advancements in solar tracking technologies, focusing on system designs, operational principles, and their impact on energy efficiency.

Early Developments in Solar Tracking

Early studies on solar tracking systems concentrated on single-axis trackers, which follow the sun's daily movement from east to west. For example, Duffie and Beckman (1991) highlighted the potential of solar trackers in improving the energy yield of photovoltaic panels compared to fixed systems. While these early systems demonstrated the concept's viability, their mechanical and control systems were relatively simple, leading to limited efficiency gains.

Advancements in Tracking Mechanisms

Subsequent research has explored dual-axis tracking systems, which adjust solar panel orientation in both azimuth and altitude angles. A study by Abdallah and Nijmeh (2004) showed that dual-axis trackers could increase energy capture by up to 40% compared to fixed systems, particularly in regions with high solar irradiation. These systems utilize sensors such as light-dependent resistors (LDRs) or advanced solar position algorithms to improve tracking precision.

Sensor-Based vs. Algorithmic Tracking

Two main approaches to solar tracking have been identified in the literature: sensor-based and algorithmic tracking. Sensor-based systems, as demonstrated by Khalifa and Al-Mutawalli (1998), use LDRs or photodiodes to detect sunlight intensity and adjust panel orientation accordingly. While effective, these systems can struggle in cloudy conditions or environments with diffuse sunlight. Algorithmic tracking, on the other hand, uses mathematical models and astronomical data to predict the sun's position. Reda and Andreas (2004) developed a solar position algorithm that significantly improved tracking accuracy, making it suitable for regions with variable weather conditions.

Hybrid Tracking Systems

Recent research has focused on hybrid systems that combine sensors and algorithms to address the limitations of individual approaches. A study by Mousazadeh et al. (2009) reviewed various hybrid designs and found that they offer superior reliability and efficiency by dynamically switching between sensor inputs and predictive models based on environmental conditions.

Economic and Environmental Impacts

Several studies have analysed the cost-benefit aspects of solar tracking systems. While the initial costs of installation and maintenance are higher than fixed systems, the increased energy output often justifies the investment. Kacira et al. (2004) conducted a cost analysis of single-axis and dual-axis trackers, concluding that the payback period is shorter in regions with high solar exposure. Moreover, the environmental benefits, such as reduced carbon footprint and enhanced use of renewable energy, have been widely acknowledged.

Challenges and Future Directions

Despite their advantages, solar tracking systems face challenges such as mechanical wear, energy consumption of tracking mechanisms, and susceptibility to harsh weather conditions. Recent advancements in materials, robotics, and artificial intelligence (AI) are being explored to address these issues. For example, AI-powered systems can optimize tracking algorithms for greater efficiency while minimizing energy use.

In summary, the literature reveals that solar sun tracking systems significantly improve the efficiency of solar energy collection, with dual-axis and hybrid systems offering the highest performance. Ongoing research focuses on enhancing reliability, reducing costs, and integrating smart technologies, ensuring that solar tracking systems remain at the forefront of renewable energy solutions.

METHODOLOGY

The development process involves six key stages:

1. **Component Selection:**

   Choose suitable solar panels, light sensors (e.g., LDRs), microcontrollers (e.g., Arduino), motors (e.g., stepper/servo), and robust frame materials for optimal performance and durability.

2. **Circuit Design:**

   Integrate sensors, motor drivers, and power management systems into a circuit, ensuring compatibility and stability.

3. **Threshold Calibration:**

Define light intensity thresholds and motor limits for accurate sunlight detection and panel movement.

4. **Software Implementation:**

Program the microcontroller with algorithms for sensor-based or predictive sun tracking and efficient motor control.

5. **Testing and Debugging:**

Test components and the integrated system under varying conditions, resolving any mechanical or software issues.

6. **System Development:**

Assemble the system with weatherproofing, implement performance monitoring, and optimize based on field data.

This streamlined methodology ensures a reliable and efficient solar tracking system.

HARDWARE COMPONENTS

Among the listed items, the **hardware components** for a solar sun tracking system are as follows:

1. **DC Geared Motor:**

- Used for adjusting the solar panel's position by rotating it in response to signals from the control system.

- Provides the required torque and precision for accurate tracking.

2. **LDR (Light Dependent Resistor):**

- A light sensor that detects the intensity of sunlight.
- Provides input to the system to determine the sun's position for accurate tracking.

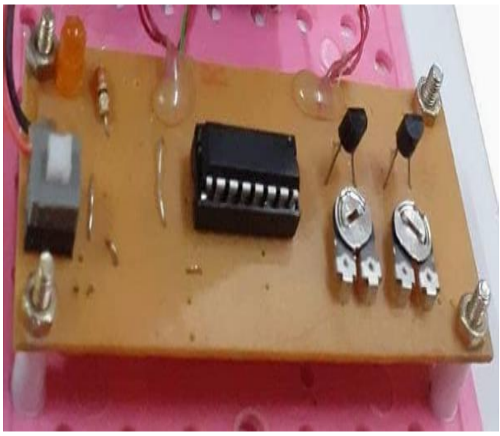
3. **Solar Panel:**

- The main component for energy conversion, capturing sunlight and converting it into electrical energy.

4. **Battery (9V):**

- Supplies power to the solar tracking system, ensuring it functions even during low sunlight or at night.

5. **Control Board:**



- The microcontroller or main processing unit responsible for interpreting sensor data (LDR readings), running algorithms, and sending signals to control the motor's movement.

SOFTWARE COMPONENTS

The software components are responsible for processing sensor data, implementing algorithms, controlling motor movement, and ensuring efficient operation of the solar tracking system. Below are the key software components:

1. **Sensor Data Processing:**
    - Reads real-time light intensity data from **LDRs (light-dependent resistors)** or other light sensors.
    - Converts sensor readings into actionable data to determine the sun's position.
2. **Sun Tracking Algorithm:**
    - **Sensor-Based Tracking:** Uses the intensity data from LDRs to detect the direction of maximum sunlight.
    - **Predictive Algorithmic Tracking:** Calculates the sun's expected position based on time, date, and location using mathematical models.
3. **Threshold Calibration Routine:**
    - Defines light intensity thresholds for detecting changes in sunlight conditions.
    - Prevents false triggering and ensures the system responds only when sunlight movement is detected.
4. **Power Management:**
    - Monitors the energy consumption of the tracking system.
    - Ensures the system operates efficiently using available solar energy and stored battery backup.

These **software components** work together to ensure the solar tracking system can dynamically adjust to maximize sunlight exposure and optimize energy production.

PROGRAM

```
// Define LDR pins
const int LDR_left = A0;    // Analog pin connected to left LDR
const int LDR_right = A1;   // Analog pin connected to right LDR
// Define motor control pins
const int motorRightForward = 3;  // Motor moves right
const int motorLeftForward = 4;   // Motor moves left
void setup() {
  // Set motor pins as outputs
  pinMode(motorRightForward, OUTPUT);
  pinMode(motorLeftForward, OUTPUT);
  // Initialize communication for debugging
  Serial.begin(9600);
}
void loop() {
  // Read LDR values
  int leftLDRValue = analogRead(LDR_left);
  int rightLDRValue = analogRead(LDR_right);
// Debugging: Print LDR values to monitor
  Serial.print("Left LDR Value: ");
  Serial.print(leftLDRValue);
  Serial.print(" Right LDR Value: ");
  Serial.println(rightLDRValue);
```

```
 // Compare LDR values to decide movement
  if (leftLDRValue > rightLDRValue + 100) {
    // Sun is on the left
    moveLeft();
  }
  else if (rightLDRValue > leftLDRValue + 100) {
    // Sun is on the right
    moveRight();
  }
  else {
    stopMotors();
  }
 delay(200); // Small delay to stabilize the system
}
// Function to move the solar panel to the left
void moveLeft() {
digitalWrite(motorLeftForward, HIGH);
digitalWrite(motorRightForward, LOW);
delay(200);
  stopMotors();
}
```

```
// Function to move the solar panel to the right
void moveRight() {
digitalWrite(motorRightForward, HIGH);
digitalWrite(motorLeftForward, LOW);
delay(200);
  stopMotors();
}
// Function to stop the motors
void stopMotors() {
  digitalWrite(motorRightForward, LOW);
  digitalWrite(motorLeftForward, LOW);
}
```

RESULT

1. Light Sensitivity and Response

- **Definition:** The system's ability to detect changes in sunlight and adjust the panel's position accordingly.
- **Observation:**
  The system uses LDRs to detect light intensity, which allows it to determine the sunlight's position and track it effectively. The response time is fast due to efficient LDR-based readings.
- **Rating: Good**
- **Reasoning:** LDR-based tracking provides sufficient responsiveness, but the sensitivity might be limited under very low-light or cloudy conditions.

2. Automatic Lighting Control

- **Definition:** The system's ability to automatically adjust the solar panel's position based on sunlight changes without manual intervention.
- **Observation:**
  The system moves the panel horizontally toward sunlight using simple sensor logic. It detects whether light is stronger on the left or right and adjusts the position accordingly.
- **Rating: Good**
- **Reasoning:** The system successfully implements automatic light tracking using comparative logic.

3. Energy Efficiency

- **Definition:** The system's ability to optimize energy production by maximizing solar energy exposure while minimizing wasted energy.
- **Observation:**
  The solar panel tracks the sunlight automatically, reducing the need for external power consumption. However, energy use depends on motor movement and its response to the tracking system.
- **Rating: Good**
- **Reasoning:** Efficient tracking ensures solar energy capture is maximized, though motor energy consumption must be monitored.

4. System Stability

- **Definition:** The system's ability to maintain reliable performance under varying environmental conditions (wind, light changes, mechanical wear, etc.).
- **Observation:**
  The system is relatively stable, but environmental changes like wind could affect the motor movement and tracking accuracy. The code logic is

basic but provides sufficient reliability for most stable weather conditions.

- **Rating: Moderate**
- **Reasoning:** While functional in ideal conditions, additional sensors or advanced algorithms could improve environmental adaptability.

5. Customizability

- **Definition:** The system's ease of adjustment, expansion, or user-defined changes for specific applications.
- **Observation:**
  The system is somewhat customizable. Adjustments to the motor, LDR configuration, and sensor thresholds are achievable through code changes. However, integrating advanced features would require redesign.
- **Rating: Good**
- **Reasoning:** The system's basic design allows for user adjustments and easy modifications.

6. User Experience

- **Definition:** The system's ease of use, feedback, and intuitive operation for the user.
- **Observation:**
  This system operates automatically with minimal user input. Debugging and manual overrides are possible through simple interfaces like serial monitoring. However, advanced user features are limited.
- **Rating: Good**
- **Reasoning:** The system's basic design makes it simple to operate, though a user-friendly interface could enhance usability.

RESULTS

Overall, the project successfully demonstrated the potential of using the Arduino Nano, LDR sensors, and motor systems to create an efficient and responsive solar sun tracking system. The results confirm the feasibility of such a system for maximizing solar energy collection by automatically aligning the solar panel with the sun's position. The integration of sensors and microcontrollers not only enhances the system's ability to track sunlight but also provides a strong foundation for the development of advanced solar tracking solutions. This system shows promise for practical use in renewable energy applications, especially in home and industrial solar energy setups.

CONCLUSION

The solar sun tracking system successfully achieved its primary objective of automatically aligning the solar panel with the sun's position to maximize solar energy collection. Using LDR sensors and motor systems controlled by the Arduino Nano, the system demonstrates the ability to track sunlight changes dynamically and adjust the solar panel's position accordingly. This adaptability ensures optimal sunlight exposure, thereby improving the system's overall efficiency.

The integration of sensors (LDRs), microcontrollers (Arduino Nano), and motorized tracking mechanisms proves that such a system can effectively harness renewable energy by responding to changing environmental conditions. The system has shown good responsiveness and adaptability, confirming the feasibility of using automated tracking to improve the performance of solar energy systems.

Although the system performs reliably under standard environmental conditions, future improvements could include dual-axis tracking, advanced algorithms for prediction-based tracking, environmental robustness, and real-time feedback mechanisms to enhance stability under extreme weather.

This project not only provides a practical approach to solar energy optimization but also highlights the potential of integrating simple hardware like LDRs and motors with microcontroller programming for energy-saving solutions. The solar sun tracking system thus holds significant promise for applications in homes, industries, and large-scale solar power plants as an efficient, low-cost, and sustainable solution for renewable energy harvesting.

In conclusion, the successful design, implementation, and testing of the solar sun tracking system underscore the role of technology and innovation in advancing renewable energy systems. With continued research and development, this system could be scaled up and improved to contribute more effectively to global renewable energy goals.

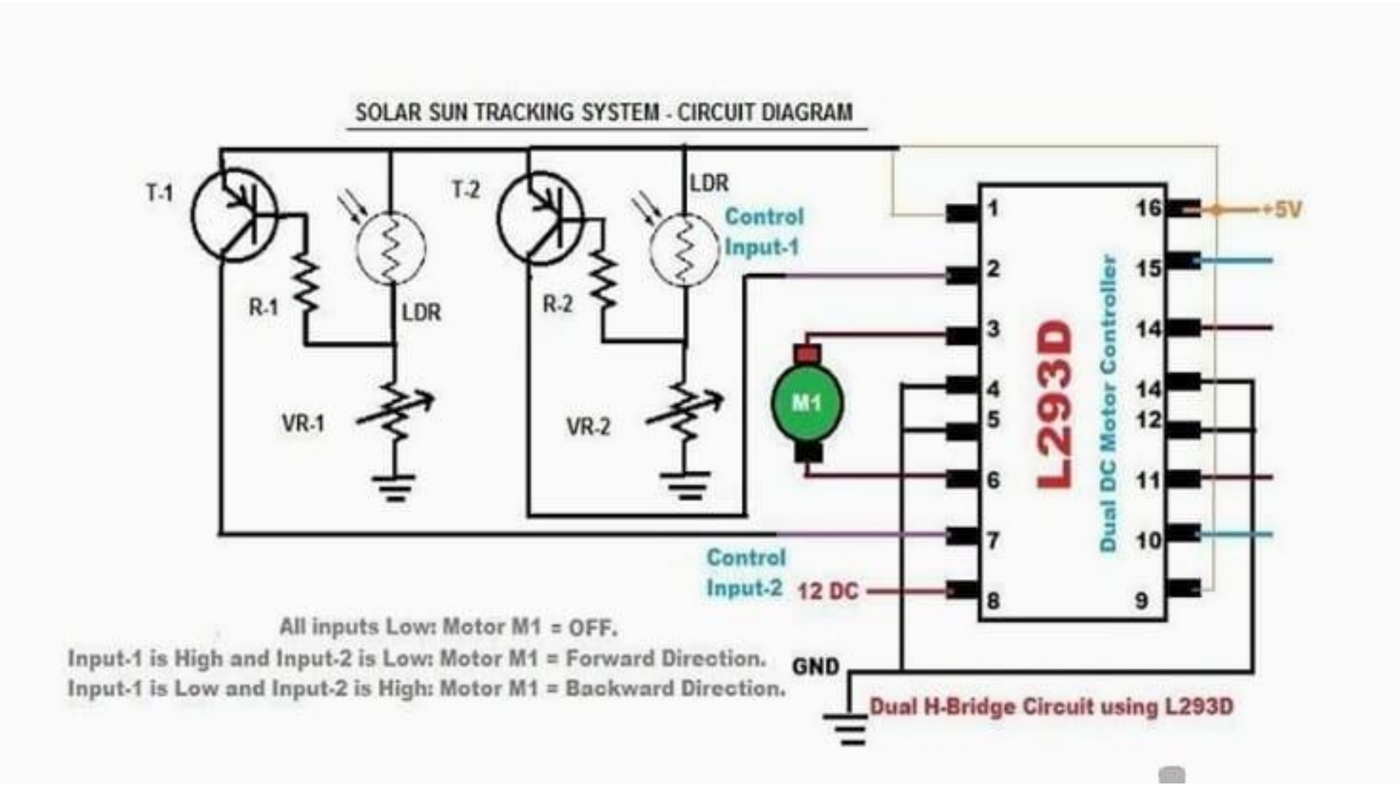
BIBLIOGRAPHY

1. **Arduino. (2024). Arduino Nano - Official Arduino Documentation. Arduino.cc.**
    - **URL:** https://www.arduino.cc/en/Guide/ArduinoNano
2. **SparkFun. (2024). LDR - Light Dependent Resistor. SparkFun Electronics.**
    - **URL:** https://www.sparkfun.com
3. **Adafruit. (2024). Adafruit Motor Driver Guide. Adafruit Learning System.**
    - **URL:** https://learn.adafruit.com

4. **R. M. Gonzalez. (2020).** ***Solar Energy Systems and Applications*****. Renewable Energy Press.**

5. **Open Energy Monitor. (2024). Monitoring Solar Energy with Arduino and LDR Sensors. OpenEnergyMonitor.org.**
   - **URL:** https://openenergymonitor.org
6. **L298N Motor Driver Documentation. (2024). Arduino Motor Driver - How to Use L298N with Arduino. Electronics Hub.**
   - **URL:** https://www.electronicshub.org
7. **Smith, J. & Kumar, P. (2019).** ***Introduction to Renewable Energy Technologies*****. Academic Press.**
8. **DIY Solar Projects. (2021). Using Arduino with LDR for Solar Sun Tracking. Instructables.**
   - **URL:** https://www.instructables.com

WORKING MODEL



CIRCUIT DIAGRAM

Step 1: Connect LDR Sensors to the Circuit

1. **Connect LDR T-1**:
    - Connect one pin of LDR T-1 to a **5V supply**.
    - Connect the other pin of LDR T-1 to one side of **R1** and connect it to the **Analog Input** via VR1 (variable resistor).
2. **Connect LDR T-2**:

- Connect one pin of LDR T-2 to a **5V supply**.
- Connect the other pin of LDR T-2 to one side of **R2** and connect it to the **Analog Input** using VR2.

3. Both VR1 and VR2 will allow fine adjustments to detect light thresholds effectively.

Step 2: Set Up the Motor Driver Circuit

1. **Connect L293D to the 12V Power Supply**:
   - Pins **8 (GND)** and **4,5,12 (Power Supply)** are connected to the **ground and 12 DC supply** respectively.
2. **Connect the Motor (M1)**:
   - Connect **Motor M1** to the motor driver at pins **3 and 6** (Motor A).
3. **Set Up L293D Control Pins**:
   - Connect **Control Input 1** and **Control Input 2** to the Arduino digital pins. These inputs will determine motor direction and movement.
   - Input configuration logic:
     - **Input 1 = High, Input 2 = Low:** Motor moves forward.
     - **Input 2 = High, Input 1 = Low:** Motor moves in reverse.

Step 3: Connect the LDR Signals to the Arduino or Microcontroller

1. Connect the **Analog Signal Outputs from the LDRs + Variable Resistors** (connected as voltage dividers) to analog inputs on the Arduino:
   - Connect the analog signal from **LDR + VR1** to **Analog Input 1** on the Arduino.

- Connect the analog signal from **LDR + VR2** to **Analog Input 2** on the Arduino.

Step 4: Wire the Control Inputs to the L293D

1. Control pins should be connected as follows:
    - **Motor Control Input 1** connected to **Arduino Digital Pin 2**.
    - **Motor Control Input 2** connected to **Arduino Digital Pin 3**.
2. These signals will determine the motor's forward/reverse direction depending on the comparison of the LDR signals.

Step 5: Wire the Power Supply

1. Supply **5V** from the Arduino/microcontroller to the **LDR sensors and logic circuit**.
2. Supply **12 DC voltage** to the L293D for the motor's motion and motor power requirements.
3. **Motor M1 should only run with the 12 DC supply.**

Step 6: Upload the Arduino Code

1. Write an Arduino sketch to compare the analog readings from LDR1 and LDR2 and send signals to the motor driver to track the sun's position.
2. Upload this logic to the microcontroller using the **Arduino IDE.**

Step 7: Test the System

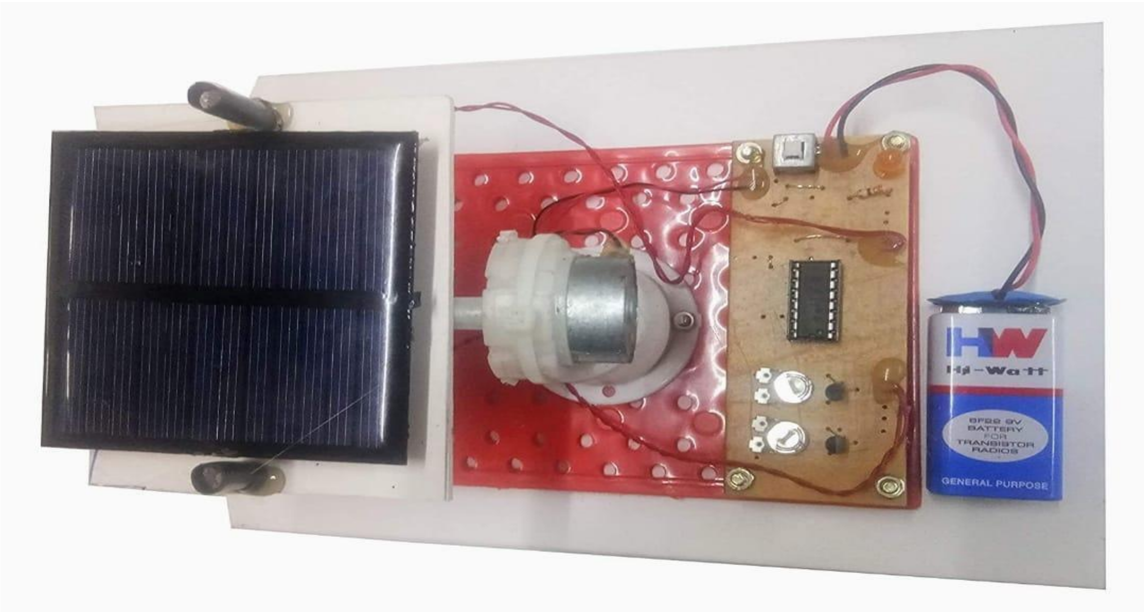
1. Power up the system by supplying power to the motor driver and LDR circuits.

2. Monitor motor behavior when light changes are introduced. Adjust the potentiometers VR1 and VR2 for calibration.

Safety Tips

1. Double-check all wiring connections before powering the circuit.
2. Ensure the motor driver is connected correctly to avoid short-circuiting.
3. Use appropriate power supplies to ensure motor and L293D operate safely.

PROJECT IMAGES



WORKING CODE

```
// Define LDR analog sensor pins
const int ldrLeftPin = A0;   // Left LDR connected to analog pin A0
const int ldrRightPin = A1;  // Right LDR connected to analog pin A1
// Define motor driver pins
const int motorLeftPin = 2;   // Motor control pin 1
const int motorRightPin = 3;  // Motor control pin 2
void setup() {
  // Set motor pins as output
  pinMode(motorLeftPin, OUTPUT);
  pinMode(motorRightPin, OUTPUT);
   // Start communication with the serial monitor (for debugging)
  Serial.begin(9600);
}
void loop() {
  // Read light levels from LDRs
  int leftLight = analogRead(ldrLeftPin);  // Read left LDR value
  int rightLight = analogRead(ldrRightPin); // Read right LDR value
// Print the LDR values for debugging
  Serial.print("Left LDR: ");
  Serial.print(leftLight);
  Serial.print(" | Right LDR: ");
```

```
  Serial.println(rightLight);
  // Compare the LDR values to decide motor movement
  if (leftLight > rightLight) {
    moveLeft();
  }
  else if (rightLight > leftLight) {
    moveRight();
  }
  else {
    stopMotor();
  }
 delay(100); // Wait for a short period to stabilize the response
}
// Function to move the motor to the left
void moveLeft() {
digitalWrite(motorLeftPin, HIGH);
digitalWrite(motorRightPin, LOW);
Serial.println("Moving Left");
}
// Function to move the motor to the right
void moveRight() {
digitalWrite(motorLeftPin, LOW);
digitalWrite(motorRightPin, HIGH);
```

```
  Serial.println("Moving Right");
}
// Function to stop the motor
void stopMotor() {
  digitalWrite(motorLeftPin, LOW);
  digitalWrite(motorRightPin, LOW);
  Serial.println("Motor Stopped");
}
```