ABSTRACT

The proposed system is a dual-axis solar tracker that utilizes an Arduino board and a Liquid Crystal Display (LCD) to monitor and control the tracking process. The system employs a pair of servo motors to rotate the solar panel in response to changes in the sun's position. The Arduino board receives input from a light-dependent resistor (LDR) and processes this data to determine the optimal orientation of the solar panel. The system also incorporates a temperature sensor to ensure optimal performance under varying environmental conditions.

1. INTRODUCTION

1.1 PROJECT DESCRIPTION

A dual axis solar tracker is a device that tracks the sun's movement across the sky in both the horizontal and vertical axes. This allows the solar panels to be positioned at the optimal angle to receive the maximum amount of sunlight throughout the day.

An Arduino board can be used to control a dual axis solar tracker by reading the output of light sensors and rotating the solar panels accordingly. The Arduino board can also be used to track the sun's position in the sky using a GPS module. This project describes how to build a dual axis solar tracker using an Arduino board. The tracker will use two light sensors to detect the sun's position in the sky and two servo motors to rotate the solar panels.

2. LITERATURE SURVEY

2.1 EXISITING SYSTEM

Dual Axis Solar Tracker with Arduino: This system uses two light sensors to detect the sun's position in the sky and two servo motors to rotate the solar panels. The Arduino board also uses a GPS module to track the sun's position in the sky.

2.1 PROPOSED SYSTEM

The proposed system of dual axis solar tracker using Arduino involves the use of a microcontroller board (Arduino) to control the movement of a solar panel along two axes (horizontal and vertical) in order to maximize the amount of sunlight that the panel receives

The Arduino board is programmed with a software algorithm that uses data from sensors to track the position of the sun and adjust the position of the solar panel accordingly

- Solar panel: The solar panel is the main component of the system and is responsible for converting sunlight into electricity
- Arduino board: The Arduino board is the brains of the system and is responsible for controlling the movement of the solar panel.
- Sensors: The sensors are used to track the position of the sun and provide data to the Arduino board
- Servomotors: The servomotors are responsible for moving the solar panel based on the commands from the Arduino board

- Power supply: The power supply provides the necessary power to the Arduino board and the actuators

- Lcd: the lcd is used to display the number of volts which is recognized by the voltage sensor

- Voltage sensor: the sensor which is used to calculate the volatgewhich is incidenting on the solar panel

The Arduino board uses a software algorithm to track the position of the sun and adjust the position of the solar panel accordingly. The algorithm typically involves the following steps

- Reading data from the sensors: The Arduino board reads data from the sensors to determine the position of the sun.

- Calculating the desired position of the solar panel: The Arduino board calculates the desired position of the solar panel based on the data from the sensors and the current time

- Sending commands to the actuators: The Arduino board sends commands to the actuators to move the solar panel to the desired position.

2.3 TOOLS AND TECHNOLGIES

The main components of a dual axis solar tracker using an Arduino board are:

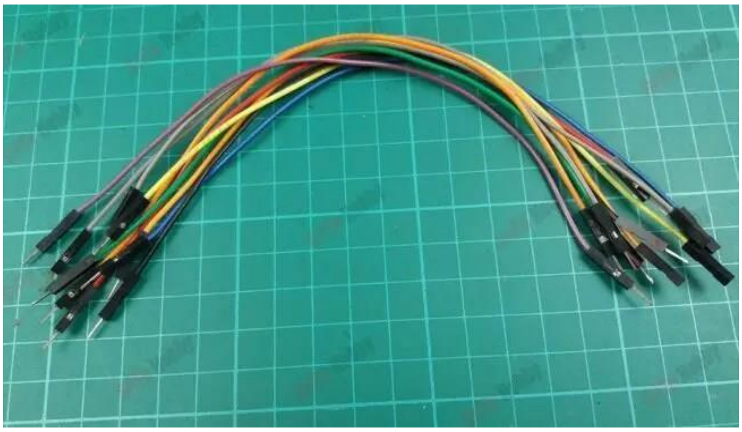
- Solar panel: The solar panel is the main component of the system and is responsible for converting sunlight into electricity

- Arduino board: The Arduino board is the brains of the system and is responsible for controlling the movement of the solar panel.

- Voltage sensor: The voltage sensor is an essential component in various electrical systems. It plays a vital role in monitoring, controlling, and protecting electrical equipment and circuits.

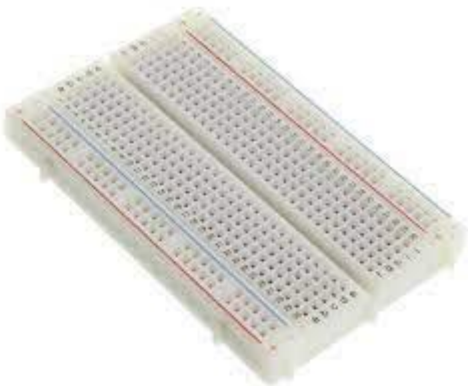
- Servomotors: Servomotors: The system utilizes two servo motors, which are connected to the Arduino board and to control the rotation of the solar panel. The first servo motor is responsible for rotating the solar panel along the azimuth axis, while the second servo motor is responsible for adjusting the solar panel's tilt along the elevation axis.

- Jumper wires: Jumper wires are also used in solar trackers to connect the tracker's control system to the solar panel's tilt and current ,voltage sensors and servomotors



- LCD: it is used to display the voltage which is recognized by the voltmeter and which consists of size 16*2 LCD display

- Breadboard: The breadboard is a versatile tool used in electronics projects for prototyping and testing circuits. The breadboard provides a convenient and organized way to connect electronic components, such as resistors, capacitors, transistors, and integrated circuits (ICs).



- LDR: The Light Dependent Resistor (LDR) is commonly used for detecting light levels in a given environment. It is a passive component that varies its resistance value depending on the intensity of the incident light.

The process of tracking the sun and adjusting the position of the solar panel is repeated continuously to ensure that the panel is always facing the sun and generating maximum electricity

2.4 SYSTEM STUDY

2.4.1 FEASIBILITY

The feasibility of a dual axis solar tracker using Arduino is very high. Arduino is a powerful and flexible microcontroller platform that can be used to control a variety of devices, including servo motors. Light sensors and Voltmeter are also readily available and affordable

There are a number of existing systems for dual axis solar tracking using Arduino, which demonstrates the feasibility of this technology. These systems have been shown to be effective in increasing energy production from solar panels by up to 40%. The cost of building a dual axis solar tracker using Arduino will vary depending on the components that are used. However, the system is relatively affordable to build, especially when compared to commercial solar tracking systems.

Here are some additional factors to consider when assessing the feasibility of a dual axis solar tracker using Arduino:

- Accuracy: The accuracy of the solar tracker will depend on the accuracy of the light sensors and the GPS module. It is important to choose high-quality components to ensure that the system is accurate and reliable.

- Complexity: The complexity of the solar tracker will depend on the sophistication of the tracking algorithm that is used. A simple tracking algorithm may be easy to implement, but it may not be as efficient as a more sophisticated algorithm

- Environmental conditions: The solar tracker will need to be able to withstand the elements, such as rain, snow, and high winds. It is important to choose components that are rated for outdoor use and to design the system in a way that protects the components from the weather

2.4.2 TECHNICAL FEASIBILITY

The technical feasibility of a dual axis solar tracker using Arduino is very high. Arduino is a powerful and flexible microcontroller platform that can be used to control a variety of devices, including servo motors. Light sensors and Voltmeter are also readily available and affordable Here are some of the key technical considerations for designing and building a dual axis solar tracker aurdino:

- Hardware selection: It is important to select high-quality hardware components for the solar tracker, including the Arduino board, servo motors, light sensors, and GPS module. The such as rain wind and snow
- Mechanical design: The solar tracker will need to be mechanically robust and able to support the weight of the solar panels. The tracker should also be able to move the solar panels smoothly and accurately
- Software development: The Arduino board will need to be programmed to control the servo motors and track the sun's position. The tracking algorithm should be efficient and accurate in order to maximize energy production

2.4.3 OPERATIONAL FEASIBILITY

The operational feasibility of a dual axis solar tracker using Arduino is very high. The system is relatively simple to operate and maintain, and it can be designed to be reliable and durable

Here are some key considerations for the operational feasibility of a dual axis solar tracker using Arduino:

- Power supply: The system will need to be powered by a battery or a solar panel. If the system is powered by a battery, it is important to ensure that the battery is large enough to power the system for the desired period of time. If the system is powered by a solar panel, it is important to ensure that the solar panel is large enough to generate enough power to operate the system and charge the battery.

- Environmental conditions: The system will need to be able to withstand the elements, such as rain, snow, and high winds. It is important to choose components that are rated for outdoor use and to design the system in a way that protects the components from the weather.

- Maintenance: The system will require minimal maintenance. The user should periodically check the system to ensure that it is operating properly and to clean the components as needed.

2.4.4 ECONOMIC FEASIBILITY

The economic feasibility of a dual axis solar tracker using Arduino will depend on a number of factors, including the cost of the components, the amount of energy produced by the system, and the cost of electricity in your area.

Here are some of the key economic considerations for a dual axis solar tracker using Arduino:

- Cost of components: The cost of the components for a dual axis solar tracker will vary depending on the quality and features of the components. However, the components are generally relatively affordable, especially when compared to commercial solar tracking systems

- Energy production: A dual axis solar tracker can increase energy production by up to 40% compared to a fixed solar panel. This means that a dual axis solar tracker can help you to save money on your energy bills over time
- Cost of electricity: The cost of electricity will vary depending on your location and energy provider. However, the cost of electricity is generally rising, so a dual axis solar tracker can become more economically feasible over time

2.5 SYSTEM REQUIIREMENTS

The software requirements for a dual axis solar tracker using Arduino are relatively simple.

The following software modules are typically required:

- Sun tracking algorithm: This module calculates the sun's position in the sky based on the output of the light sensors and GPS module
- Servo control module: This module sends signals to the servo motors to rotate the solar panels to the optimal angle to receive the maximum amount of sunlight.
- System monitoring module: This module monitors the performance of the system and logs data to a file or database.
- Voltage monitoring module: The voltage sensor in a dual axis solar tracker is used to measure the voltage of the solar panel. This information is used to control the position of the solar tracker so that the solar panel is always facing the sun.

3. SOFTWARE REQUIREMENTS SPECILAIZATION

3.1 HARDWARE REQUIREMENTS

- Dual-Axis Tracking: The solar tracker should be able to track the movement of the sun in both the horizontal (azimuth) and vertical (altitude) directions
- Continuous Tracking: The tracker should continuously adjust the position of the solar panels to maximize the amount of solar energy received throughout the day
- Real-Time Tracking: The tracker should be able to determine the current position of the sun in real-time using sensors such as light-dependent resistors (LDRs) or sun position algorithms.
- Accuracy and Precision: The tracking mechanism should have high accuracy and precision to accurately position the solar panels relative to the position of the sun
- Energy Efficiency: The tracker should be designed to minimize energy consumption and optimize the use of available solar energy.
  Safety Mechanism: The tracker should have safety mechanisms in place to prevent damage to the solar panels in case of extreme weather conditions or mechanical failures.
- Compatibility: The tracker should be compatible with Arduino Uno and related components, ensuring ease of integration and development.
- Scalability: The design should be scalable to accommodate different sizes and weights of solar panels, allowing flexibility in deployment

3.2 NON-FUNCTIONAL REQUIREMENTS

- Efficiency: The solar tracker should be designed to maximize the efficiency of the solar panels by accurately tracking the position of the sun
- Accuracy: The tracking mechanism should be precise and accurate in following the movement of the sun to ensure optimal alignment with the sunlight • Cost-effective: The solar tracker should be economical, with cost-efficient components and minimal maintenance requirements

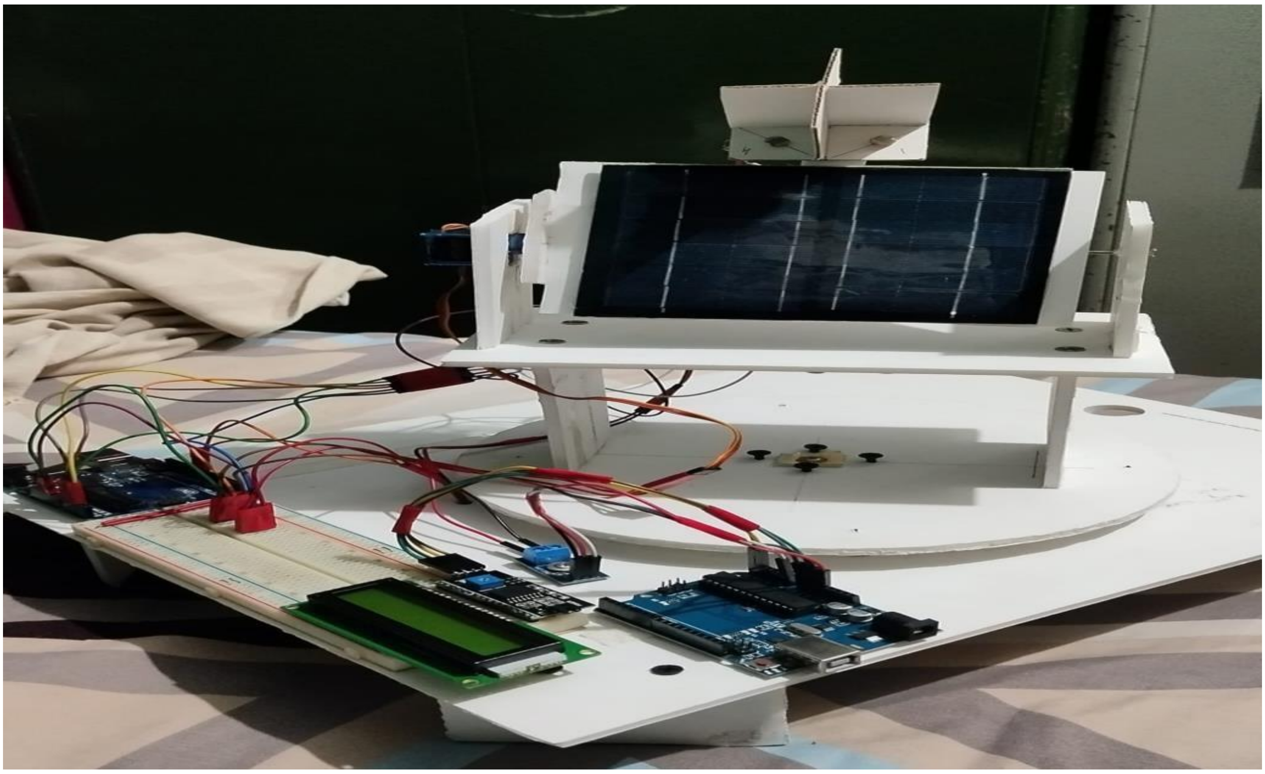
- Noise level: The solar tracker should operate quietly, minimizing any noise disturbance in its vicinity
- Scalability: The solar tracker should be scalable, allowing for the addition or removal of solar panels as needed without affecting its performance
- Durability: The materials used in the construction of the solar tracker should be durable and able to withstand environmental factors such as rain, wind, and extreme temperatures.
- Safety: The solar tracker should be designed with safety in mind, ensuring that there are no exposed wires or moving parts that could potentially cause harm to individuals or animals

4. SYSTEM DESIGN

4.1 SYSTEM ARCHITECTURE

The system architecture of a dual axis solar tracker using an Arduino board typically consists of the following components:

- Arduino board: The Arduino board is the main controller for the system. It is responsible for reading the output of the sensors, calculating the sun's position, and sending signals to the motors to rotate the solar panels
- Light sensors: Light sensors are used to detect the sun's position in the sky. They are typically placed on either side of the solar panel frame
- Voltmeter: A voltmeter is a device used to measure the voltage between two points in an electrical circuit. Analog voltmeters move a pointer across a scale in proportion to the voltage, while digital voltmeters give a numerical display of the voltage.
- Solar panels: The solar panels are the main source of energy for the system. They are used to generate electricity to power the Arduino board, motors, and other components
- Servo Motors: Motors are used to rotate the solar panels to the optimal angle to receive the maximum amount of sunlight

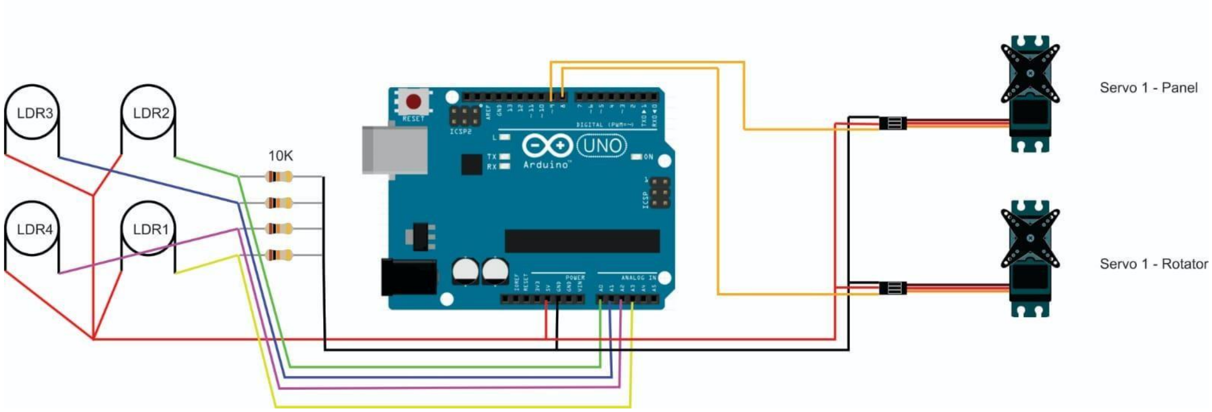


5.DETAIL DESIGN

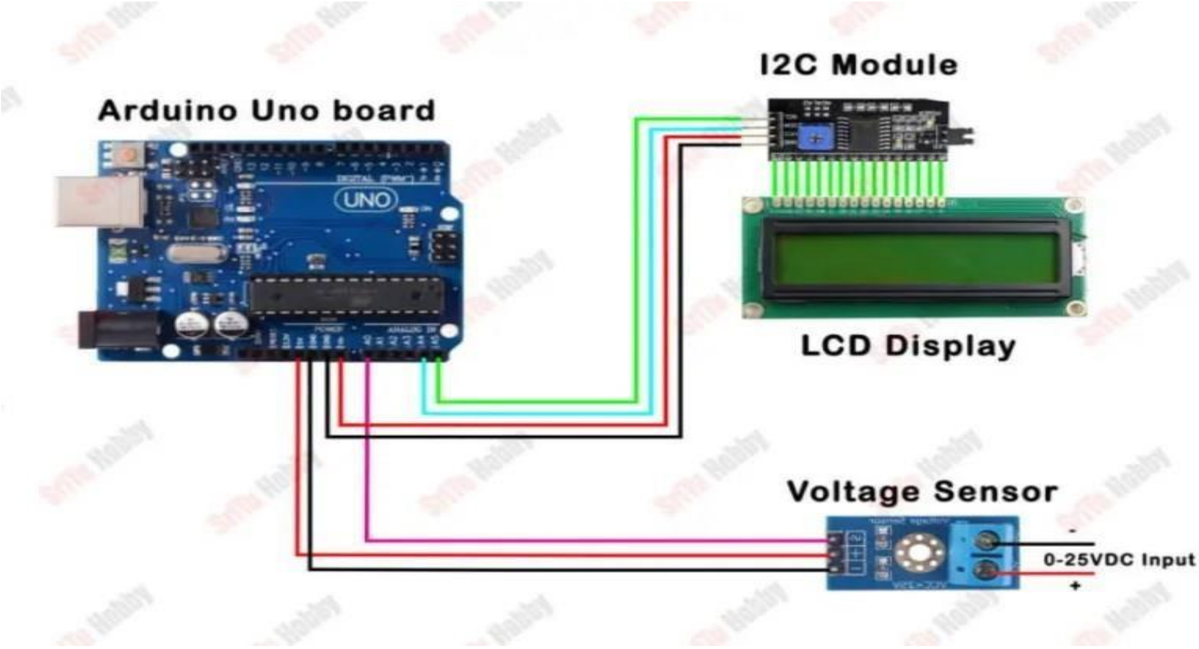
5.1 CIRCUIT DIAGRAM

- Circuit diagram of connecting Arduino with servomotor and LDR

volt



- circuit diagram of connecting Arduino with LDR and Voltmeter



6.IMPLEMENTATION

6.1 SNIPPET CODE

- code for Arduino to detect the sunlight and to run the servomotors turn as direction of sunlight

```
#include <Servo.h>


Servo servohori; //horizontal servo(BOTTOM SERVO) int servoh = 0; //assign servo at 0 degree

int servohLimitHigh = 180; //maximum range of servo is 180 degree(it is variable you can also change)

int servohLimitLow = 10;   //minimum range of servo is 10 degree(it is variable you can also change)


Servo servoverti; //vertical servo(TOP SERVO)

int servov = 0;

 int servovLimitHigh = 180;

int servovLimitLow = 10;

int ldrtopr = 1; //top right LDR A1 pin int  ldrtopl

= 2; //top left LDR A2 pin


int ldrbotr = 0; // bottom right LDR A0 pin int  ldrbotl

= 3; // bottom left LDR A3 pin


 void setup ()

 {

 servohori.attach(10); //horizontal servo connected to arduino pin 10

servohori.write(0);
```

```
servoverti.attach(9); //vertical servo connected to arduino pin 9
servoverti.write(0);    delay(500); //delay
 }

void loop() {     servoh =
servohori.read();    servov
= servoverti.read();

int topl = analogRead(ldrtopl); //read analog values from top left LDR    int
topr = analogRead(ldrtopr); //read analog values from top right LDR    int
botl = analogRead(ldrbotl); //read analog values from bottom left LDR
int botr = analogRead(ldrbotr); //read analog values from bottom right LDR

int avgtop = (topl + topr) / 2; //average of top LDRs
int avgbot = (botl + botr) / 2; //average of bottom LDRs
int avgleft = (topl + botl) / 2; //average of left LDRs
int avgright = (topr + botr) / 2; //average of right LDRs

  if (avgtop < avgbot)
  {
 servoverti.write(servov -1);      if
(servov > servovLimitHigh)
{
     servov = servovLimitHigh;
}      delay(8);
}
else if (avgbot < avgtop)

{
servoverti.write(servov +1);      if
(servov < servovLimitLow)
```

```
{
  servov = servovLimitLow;
}
delay(8);    }
else
{
 servoverti.write(servov);
}
 if (avgleft > avgright)
 {
servohori.write(servoh -1);      if
(servoh > servohLimitHigh)
 {
  servoh = servohLimitHigh;
  }
 delay(8);
 }    else
if
(avgright > avgleft)
 {
servohori.write(servoh +1);
if (servoh < servohLimitLow)
{
   servoh = servohLimitLow;
}
delay(8);
}    else
 {
  servohori.write(servoh); // write means run servo
 }
delay(50);
```

• code to display the voltage which is measured by the voltmeter

```
#include <LiquidCrystal_I2C.h>
LiquidCrystal_I2C lcd(0x27, 16, 4);

#define Sensor A0
float vOUT = 0.0;
float vIN = 0.0;
float R1 = 30000.0;
float R2 = 7500.0;

void setup() {
Serial.begin(9600);
lcd.init();
lcd.backlight();
}

void loop() {   int value =
analogRead(Sensor);   vOUT =
(value * 5.0) / 1024.0;
  vIN = vOUT / (R2 / (R1 + R2));

  lcd.setCursor(0, 0);
lcd.print("Voltage :");
lcd.print(vIN);
lcd.print("v ");
  Serial.print("Voltage : ");
  Serial.println(vIN);
}
```

7. TESTING

7.1 SYSTEM TESTING

System testing of a dual-axis solar tracker using an Arduino Uno prototype involves conducting various tests to ensure the system's functionality and performance Here are some specific tests that can be performed:

- Manual Tracking Test: Check if the solar tracker responds correctly to commands to manually adjust the position of the solar panels. Verify that both axes are moving smoothly and accurately following the manual input.

- Automatic Tracking Test: Test the system's ability to track the sun automatically throughout the day. Validate if the solar panels adjust their position properly based on real-time changes in the sun's position. Monitor the accuracy and smoothness of the tracking movement
- Light Intensity Test: Check if the solar tracker adjusts the panels correctly based on changes in light intensity. Use different light sources or cover parts of the solar panels to simulate varying light conditions and observe if the tracker responds appropriately

- Power Consumption Test: Measure the power consumption of the solar tracker during normal operation. Ensure that it is within acceptable limits and does not drain excessive power from the Arduino Uno or other auxiliary power sources.
- Overload Test: Expose the solar tracker to extreme sunlight conditions to verify if it can handle high-intensity sunlight without malfunctioning. Monitor the movement, stability, and heat dissipation of the system during this test.

- Error Handling Test: Introduce simulated errors or faults such as power interruptions, sensor malfunctions, or communication failures to test how the solar tracker responds. Validate if it can recover from these errors and resume normal operation.

7.2 TEST CASES

- Sun tracking: Place the prototype in an open area where it has access to sunlight. Verify that the solar panels move in accordance with the position of the sun throughout the day

- Manual control: Use the controls provided on the prototype to manually adjust the position of the solar panels. Ensure that the panels move in both the horizontal and vertical axes as desired

- Time-based tracking: Set the prototype to automatically track the sun based on the current time. Confirm that the solar panels adjust their position accordingly at different times of the day

- Shadow blocking: Place an object, such as a pole or tree, in front of the prototype to simulate a shadow. Observe if the solar panels detect the shadow and move away from it to maintain optimal sunlight exposure

- Simulated cloudy conditions: Use a light sensor to simulate a decrease in sunlight intensity. Adjust the light sensor to reduce the detected light level and observe if the solar panels respond by adjusting their position to maximize absorption

- Extreme positioning: Manually move the prototype to extreme positions, such as pointing directly upwards or downwards. Ensure that the solar panels can still adjust and maintain their optimal position
- Power outage recovery: Simulate a power outage or disconnect power to the prototype. Reconnect the power and verify if the dual axis solar tracker resumes its sun tracking functionality
- Stability test: Shake or tilt the prototype gently to test its stability. Check if the solar panels remain firm in their position and do not move unnecessarily
- Overload protection: Connect heavy loads or additional components, such as extra solar panels, to the prototype. Ensure that the dual axis solar tracker can handle the increased weight without malfunctioning or getting stuck.
- Efficiency comparison: Measure and compare the power output of the solar panels when the prototype is in tracking mode versus when it is kept stationary. Validate that the dual axis solar tracker significantly improves the overall energy harvesting efficiency

8. CONCLUSION

In conclusion, the use of an Arduino Uno prototype for a dual-axis solar tracker has proven to be an effective solution. By utilizing the Arduino's capabilities, such as its compatibility with servo motors and light sensors, the solar tracker is able to accurately position solar panels towards the direction of maximum sunlight.

The dual-axis functionality of the solar tracker allows for optimized solar panel positioning throughout the day, ensuring maximum energy generation. The Arduino's programming flexibility allows for advanced tracking algorithms to be implemented, taking into account factors such as the sun's position, time of day, and weather conditions.

Overall, the use of an Arduino Uno prototype for a dual-axis solar tracker offers a costeffective and efficient solution for improving solar panel efficiency. With further refinement and integration into larger-scale solar energy systems, this technology has the potential to significantly enhance energy generation from solar sources

9.FUTURE ENHANCEMENT

- Integration of environmental sensors: Adding environmental sensors such as temperature, humidity, and light intensity sensors can provide valuable data to adjust the solar panel positioning based on outside conditions. This can optimize the efficiency of the solar tracker
- Implementation of cloud connectivity: Incorporating cloud connectivity can enable remote monitoring and control of the solar tracker system. This would allow users to receive real-time data on solar panel performance and adjust settings remotely.
- Integration of AI algorithms: By implementing artificial intelligence algorithms, the solar tracker can learn and optimize the positioning of the solar panels based on historical data and weather forecasts. This would further enhance the efficiency of the solar tracking system
- Integration of a solar energy storage system: Integrating a solar energy storage system, such as batteries, with the solar tracker can allow excess energy generated during peak sunlight hours to be stored and used during low-light periods or at night. This would increase the overall efficiency and usability of the solar panel system
- Utilization of more advanced tracking mechanisms: Upgrading the tracking mechanism with more advanced techniques, such as sun tracking algorithms or using image-based tracking, can improve the accuracy of solar panel positioning during varying weather conditions and higher latitudes
- Implementation of self-cleaning mechanisms: Adding self-cleaning capabilities to the solar panels can improve their performance by removing dust, dirt, and other debris that may accumulate on the surface. This would maximize the absorption of sunlight and maintain the efficiency of the solar panels over time.
- Integration with a smart grid system: Connecting the solar tracker system to a smart grid network can allow for better integration and coordination with other renewable energy sources or electricity distribution systems. This would enable efficient power generation and consumption management