DESIGN OF SINGLE AXIS SOLAR TRACKER

## A MINI PROJECT REPORT

*Submitted by*

|  |  |
| --- | --- |
| **GOWTHAMRAJ S** | **420420104014** |
| **HARISH N** | **420420104016** |
| **SEDHUMADHAVAN V** | **420420104044** |

*in partial fulfillment for the award of the degree*

*of*

# BACHELOR OF ENGINEERING

*in*

COMPUTER SCIENCE AND ENGINEERING



**ADHIPARASAKTHI ENGINEERING COLLEGE MELMARUVATHUR-603 319**

# ANNA UNIVERSITY :: CHENNAI 600 025

MAY 2023

**ANNA UNIVERSITY: CHENNAI 600 025**

**BONAFIDE CERTIFICATE**

Certified that this project report titled **“ DESIGN OF SINGLE AXIS SOLAR TRACKER”** is the bonafide work of **GOWTHAMRAJ S (420420104014), HARISH N (420420104016), SEDHUMADHVAN V (420420104044)** who carried out the work under my supervision.

**SIGNATURE SIGNATURE**

**Dr. C. DHAYA, Ph.D., Mr. G. SEKAR, M.E.,**

**HEAD OF THE DEPARTMENT**

**Professor/CSE**

**SUPERVISOR**

**Assistant Professor/CSE**

Department of CSE Department of CSE

Adhiparasakthi Engineering College Adhiparasakthi Engineering College Melmaruvathur - 603 319. Melmaruvathur - 603 319.

Submitted for the (CS8611) Mini project work and viva-voce examination held on ………..at Adhiparasakthi Engineering College, Melmaruvathur.

**INTERNAL EXAMINER EXTERNAL EXAMINER**

## ACKNOWLEDGEMENT

With the divine blessings of **Goddess Adhiparasakthi**, I express my deep gratitude for the spiritual blessings of His holiness **PADMA SHRI ARULTHIRU AMMA** and the divine guidance of **THIRUMATHI AMMA** that has undoubtedly taken us to path of victory in completing this project.

The infrastructural support with all kinds of lab facilities have been a motivating factor in this completion of project work, all because of our **Correspondent Sakthi Thiru Dr. G. B. SENTHILKUMAR B.E.** with great pleasure we take this opportunity to thank him.

For the academic side the constant support from our honorable **Principal Dr. J. RAJA, Ph.D.,** as encourages us to work hard and attain this goal of completing the project.

Our sincere thanks to our **Head of the Department Dr. C. DHAYA, Ph.D.,** who has given us both moral and technical support adding experience to the job we have undertaken.

We thank our **Project Coordinator Ms.R.SRIVIDYA, M.E.,** who has led from the front questioning as at the right time and encouraging us most of the time thereby extracting quality work from us.

We sincerely thank our **Supervisor, Mr. G. SEKAR, M.E.,** who helped us in crossing in the path to our glory. We also thank other staff members, librarian, and non-teaching staff member of computer center, who have given their constant support and motivation in all our endeavor.

# ABSTRACT

The increasing demand for renewable energy source has led to a growing interest in solar power generation. Solar trackers are devices designed to maximize the efficiency of solar panels by orienting them towards the sun throughout the day. This project presents the design and implementation of a single-axis solar tracker using Arduino, a popular micro-controller platform. The objective of this project is to develop a cost-effective and reliable solution for residential or small-scale solar installations. The single-axis solar tracker system tracks the sun's movement from east to west, adjusting the solar panel's position accordingly to optimize energy generation. A comprehensive literature review explores solar energy and tracking systems, different types of solar trackers, and the advantages and limitations of single-axis solar trackers. The software design and development section delves into the programming aspect of the project. The system calibration and testing phase focuses on calibrating the tracking algorithm, evaluating tracking accuracy, and comparing energy generation with and without the solar tracker system. Results and discussions present the findings and analysis of the system's performance, including tracking accuracy, energy generation improvements, and cost-effectiveness.

**iv**

**TABLE OF CONTENTS**

|  |  |  |
| --- | --- | --- |
| **CHAPTER** | **TITLE** | **PAGE** |
| **NO** | **ACKNOWLEDGMENT** | **NO**  iii |
|  | **ABSTRACT** | iv |
|  | **LIST OF SYMBOLS** | v |
|  | **LIST OF FIGURES** | vii |
|  | **LIST OF ABBREVIATIONS** | 1 |
| **1** | **INTRODUCTION** | **2** |
|  | 1.1 DOMAIN OVERVIEW | 3 |
|  | 1.2 OVERVIEW OF THE PROJECT | 3 |
|  | 1.3 OBJECTIVE OF THE PROJECT | 3 |
| **2** | **LITERATURE SURVEY** | **5** |
|  | 2.1 REVIEW OF LITERATURE | 5 |
|  | 2.2 PROBLEM STATEMENT | 6 |
| **3** | **SYSTEM ANALYSIS** | **8** |
|  | 3.1 EXISTING SYSTEM | 8 |
|  | 3.1.1 DISADVANTAGES | 8 |
|  | 3.2 PROPOSED SYSTEM | 9 |
|  | 3.2.1 ADVANTAGES | 9 |
| **4** | **REQUIREMENT ANALYSIS** | **10** |

* 1. [FUNCTIONAL REQUIREMENTS 10](#_TOC_250019)
  2. [NON-FUNCTIONAL REQUIREMENTS](#_TOC_250018) 10
  3. HARWARE REQUIREMENTS 11

4.4 [SOFTWARE REQUIREMENTS 16](#_TOC_250017)

1. DESIGN ENGINEERING 18
   1. ARCHITECTURE DIAGRAM 18
   2. [DATA FLOW DIAGRAM 1](#_TOC_250016)9
   3. [USE CASE DIAGRAM](#_TOC_250015) 20
2. MODULES DESCRIPTION 21
   1. [IMPORTING NECESSARY LIBRARIES](#_TOC_250014) 21
   2. [DATA PREPROCESSING](#_TOC_250013) 21
   3. [MODEL DEVELOPMENT](#_TOC_250012) 21
   4. [MODEL EVALUATION](#_TOC_250011) 22
   5. [MODEL DEPLOYMENT](#_TOC_250010) 23
3. SOLAR TECHNOLOGY 24
   1. [INTRODUCTION](#_TOC_250008) 24
   2. [WORKING PRINCIPLE](#_TOC_250007) 25
   3. [COMPONENTS](#_TOC_250006) 26
   4. [CLASSIFICATION OF SOLAR](#_TOC_250005) 26
4. IMPLEMENTATION 31

8.1 CONNET THE HARDWARE COMPONENTS 31

8.2 DOWNLOAD & INSTALL ARDUINO

APPLICATION 31

8. 3 ARDUINO SETUP 31

1. SNAPSHOTS 34
   1. SAMPLE INPUT 34
   2. SAMPLE OUTPUT 35
2. CONCLUSION AND FUTURE ENHANCEMENT 38

10.2 [CONCLUSION 38](#_TOC_250002)

* 1. [FUTURE ENHANCEMENT](#_TOC_250001) 39

# LIST OF SYMBOLS

|  |
| --- |
| Class name |
| Visibility attribute Type=initial value |
| Visibility operation  (arglist):return type |

|  |  |  |
| --- | --- | --- |
| **SYMBOL NAME** | **NOTATION** | **DESCRIPTION** |
| Class |  | Class represents a collection of similar entities grouped together |
| Association |  | Association represents a static relationship between classes. |
| Use case |  | A use case is an interaction between the system and other external examination. |
| Relational |  | It is used for Additional Process Communication |
| Control flow |  | It represents the control flow between the state |
| Data process/State |  | A circle in DFD represent the vertical dimension the object  Communication |
| Object lifeline |  | An object lifeline represents the vertical dimension then object  Communication. |

|  |  |  |
| --- | --- | --- |
| **SYMBOL NAME** | **NOTATION** | **DESCRIPTION** |
| Message |  | It represents the Message exchanged |
| Actor |  | Actors are the user of the system and other external entity that react with the system |

|  |  |  |
| --- | --- | --- |
| **FIGURE**  **NO** | **FIGURE TITLE** | **PAGE**  **NO** |
| 4.3.1  4.3.2  4.3.3  4.3.4  4.3.5  5.1  5.2  5.3    7.4  8.1  8.2  8.3  9.1  9.2 | ARDUINO Uno  Servo motor  LDR  Resistor  Solar panel  System Architecture for single axis solar tracker  Data Flow diagram for single axis solar tracker  Use Case Diagram for single axis solar tracker  Classification of solar tracker  Connect the hardware components  Download & install arduino ide application Arduino setup  Sample Input  Sample Output | 11  13  13  14  15  18  19  20  26  31  31  31  35  36 |

# LIST OF ABBREVIATIONS

* Arduino - ARD
* Single-Axis Solar Tracker - SAT
* Photovoltaic - PV
* Light Dependent Resistor - LDR
* Servo Motor - SRV
* Photovoltaic Panel - PVP
* Programmable Logic Controller - PLC
* Internet of Things - IoT
* Breadboard - BB

## CHAPTER 1

## INTRODUCTION

Solar energy is rapidly gaining popularity as a clean and sustainable source of power. To harness the maximum potential of solar energy, it is crucial to optimize the performance of solar panels. One effective way to achieve this is through the use of single-axis solar trackers.A single-axis solar tracker is a device that automatically adjusts the position of solar panels to track the movement of the sun throughout the day. By continuously aligning the panels perpendicular to the sun's rays, single-axis trackers significantly enhance energy generation and improve overall system efficiency.

The main principle behind a single-axis solar tracker is its ability to tilt or rotate the solar panels along a single axis. This axis is typically aligned in an east-west direction, allowing the panels to follow the sun's path from sunrise to sunset. By tracking the sun's movement, single-axis trackers maximize the exposure of solar panels to direct sunlight, optimizing their energy output.Increased Energy Production: Single-axis trackers can boost energy generation by up to 25-35% compared to fixed solar installations. By continuously adjusting the panels' angle, trackers ensure that the panels capture the maximum amount of sunlight throughout the day.

## DOMAIN OVERVIEW

The domain of single-axis solar trackers encompasses the design, development, implementation, and operation of systems that track the sun's movement along a single axis to optimize solar energy capture. It involves various technical aspects, including mechanical engineering,control systems, sensors, and renewable energy technology .The primary goal of single-axis solar trackers is to maximize the efficiency and output of solar installations. By continuously adjusting the orientation of solar panels, these trackers ensure that the panels are aligned perpendicular to the sun's rays, maximizing the amount of sunlight absorbed.

## OVERVIEW OF THE PROJECT

The project involves the design, construction, and implementation of a single-axis solar tracker system. The goal is to enhance the energy generation efficiency of solar installations by continuously adjusting the position of solar panels to track the sun's movement.

## 1.3 OBJECTIVE OF THE PROJECT

Improve energy generation efficiency. The primary objective is to enhance the energy generation efficiency of solar installations by continuously adjusting the position of solar panels to track the sun's movement. The project aims to optimize the orientation of the panels throughout the day, ensuring they are aligned perpendicular to the sun's rays, thereby maximizing the absorption of sunlight and increasing energy production.

By implementing a single-axis solar tracker, the project aims to increase overall energy output of the solar installation compared to fixed solar panels. The tracker system will continuously adjust the panel's angle, optimizing sunlight capture and generating a higher amount of electricity. The project aims to improve the. performance of the solar panels during morning and afternoon periods when the sun's angle is lower. By precisely tracking the sun's movement, the tracker system will adjust the panels to maximize sunlight exposure during these times, further increasing energy production. Single-axis solar tracker system capable of withstanding various environmental conditions. The tracker structure, tracking mechanism, and control system should be designed to ensure long-term functionality and minimal maintenance requirements. Establish a performance monitoring and analysis framework to evaluate the effectiveness of the single-axis solar tracker system. This includes collecting data on energy generation before and after tracker implementation, analyzing the impact of tracking on energy output, and assessing the overall performance of the system.

## CHAPTER 2 LITERATURE SURVEY

A Literature review is a text of a scholarly paper, which includes the current knowledge, including substantive findings, as well as theoretical and methodological contributions to a particular topic. Literature reviews use secondary sources and do not report new or original experimental work. A literature review usually precedes the methodology and results section.

## REVIEW OF LITERATURE:

1. **Title: Single axis sun tracking solar system using IC L293D**

**Author:** Dr.S.N.Chandra shekar ; Shiva Rama Krishna :

**Year:** 2022

**Description:** The paper purpose is to optimize the amount of sunlight captured by solar panels. By tracking the sun's movement along a single axis, the system ensures that the panels are always positioned perpendicular to the sun's rays, maximizing the absorption of solar energy. This leads to increased energy capture and higher electricity generation.

1. **Title: Horizontal single axis solar tracker using arduino**

**Author:** Siti amely jumaat ,Adm afiq azlan :

**Year:** 2018

**Description:** The paper purpose aims to optimize the energy conversion efficiency of solar panels by maintaining their optimal orientation relative to the sun. Solar panels generate the maximum output when they are exposed to direct sunlight at the ideal angle. By tracking the sun's movement, the system adjusts the panels to maintain their perpendicular alignment with the sun's rays, maximizing energy conversion efficiency and overall system performance.

## Title: Efficient single axis sun tracker design for photovoltaic system applications

**Author:** J. Hussain S,Muayyad N

**Year:** 2017

**Description:** The paper purpose of the efficient single-axis sun tracker is to maximize the energy generation of photovoltaic systems. By continuously adjusting the orientation of solar panels to track the sun's movement along a single axis, the

tracker ensures that the panels are always positioned perpendicular to the sun's rays.

This maximizes the absorption of solar radiation and significantly increases the

energy output of the system.

**4. Solar tracker for maximizing power production overlapping**

## removal on the sensors of tracker

**Author:** Mst Jesmin Nahar, Rasel Sarkar

**Year:** 2021

**Description:** The paper purpose by implementing overlapping removal on the sensors, the system minimizes power losses due to shading or obstructions. When sensors are obstructed, the system adjusts the panel orientation to avoid shading and ensures that all sensors receive direct sunlight. This minimizes power losses caused by partial shading and maximizes the overall power output of the system.

**PROBLEM STATEMENT**

* The problem is that solar panels are fixed in position and cannot dynamically track the sun's movement, resulting in sub optimal energy capture and reduced efficiency. This limitation hinders the panels' ability to consistently align themselves with the sun's rays, leading to decreased power generation.
  + There is a need to develop a single-axis solar tracker using Arduino that can actively track the sun's position and adjust the panel's orientation to optimize sunlight exposure. The tracker should overcome challenges such as changing weather conditions, obstructions, and shading caused by nearby objects.

## CHAPTER 3

## 3.1 EXISTING SYSTEM ANALYSIS

The existing system for this type of tracker rotates the solar panels along a horizontal axis, typically from east to west. The panels move throughout the day to follow the path of the sun from sunrise to sunset, maximizing the amount of sunlight they receive.This tracker combines the horizontal and tilted axis movements. The solar panels rotate both horizontally and vertically, following the sun's azimuth (east-west movement) and elevation (up-down movement) angles, respectively. This design provides more precise tracking and can optimize energy generation in areas with varying solar radiation angles.The solar panels are mounted on a tilted axis that tracks the sun's movement from east to west, similar to the horizontal tracker. However, the panels are also tilted at an angle to maximize the incident sunlight, considering the sun's elevation angle throughout the day.

## 3.1.1 DISADVANTAGES

* **Limited Tracking Range:** Single Axis Solar Trackers typically move along a single axis, either from east to west or vice versa. This means they can track the sun's movement in one dimension only.
* **Higher System Complexity:** Single Axis Solar Trackers require additional mechanical components, motors, sensors, and control systems to facilitate the tracking movement. This complexity can result in higher installation and maintenance costs.
* **Tracking Accuracy and Precision**: Single Axis Solar Trackers rely on sensors and control systems to track the sun's movement accurately.

## 3.2 PROPOSED SYSTEM

* **Advanced Tracking Mechanism:** The proposed system incorporates an advanced tracking mechanism that utilizes precise sensors and control systems to accurately track the sun's movement along a single axis. This ensures optimal alignment of the solar panels with the sun's position throughout the day, maximizing energy absorption and generation.
* **High Tracking Accuracy:** The proposed system focuses on achieving high tracking accuracy to ensure that the solar panels are precisely aligned with the sun's position. This involves the use of high-quality sensors, calibration techniques, and feedback mechanisms to minimize tracking errors and deviations. Improved tracking accuracy leads to enhanced energy generation and system performance**.**
* **Robust and Reliable Design:** The proposed system emphasizes a robust and reliable design to ensure long-term operation and durability. This design approach minimizes the risk of mechanical failures and extends the lifespan of the tracking system**.**

## 3.2.1 ADVANTAGES

* **Increased Energy Generation:** Single Axis Solar Trackers significantly increase energy generation compared to fixed-tilt systems. By continuously adjusting the position of the solar panels to face the sun, trackers maximize the absorption of sunlight throughout the day.
* **Enhanced Performance in Variable Conditions:** Single Axis Solar Trackers adapt to changing solar angles throughout the day and across seasons.
* **Flexibility and Adaptability**: Single Axis Solar Trackers are versatile and

adaptable to different site conditions. They can be installed in various

locations, including residential, commercial, and utility-scale projects.

## CHAPTER 4 REQUIREMENT ANALYSIS

Requirement analysis determines the requirements of a new system. This project analyzes product and resource requirements, which is required for this successful system.

The product requirement includes input and output requirements it gives the want in terms of input to produce the required output. The resource requirements give in brief about the software and hardware that are needed to achieve the required functionality.

## FUNCTIONAL REQUIREMENTS

A Functional requirement defines the function of a system or its components. A function is described as the set of inputs, the behavior, and the outputs. Functional requirements may be calculations, technical details, data manipulation and other specific functionalities that show how a use case is to be fulfilled. They are supported by non-functionalities requirements, which impose constraints on the design or implementation.

## NON-FUNCTIONAL REQUIREMENTS

Non-Functional Requirements are requirements that specify criteria that can be used to judge the operations of a system.

Rather than specific behaviors, Non-functional requirements are often called qualities of the system. The non-functional requirements in this system are:

* + 1. The system should be accurate and efficient.
    2. The system should be able to meet all user requirements.

## HARDWARE REQUIREMENTS

The most common set of requirements defined by any operating system or software application is the physical computer resources, also known as hardware.

A hardware requirements list is often accompanied by a hardware compatibility list, especially in the case of operating systems.

The minimal hardware requirements are as follows,

1. Arduino Uno: This will serve as the microcontroller for the system.
2. Light Sensor: A light sensor, such as a photodiode or a light-dependent resistor (LDR), to measure the intensity of light.
3. Servo Motor: A servo motor that can rotate the solar panel.
4. Solar Panel: The solar panel that needs to be aligned with the sun's position.

## ADRUINO UNO

A microcontroller is a small computer on a single integrated circuit containing a processor core memory, and programmable input/output peripherals Microcontrollers are designed for embedded applications, in contrast to the microprocessors used in personal computers or other general purpose applications, in contrast to the microprocessors used in personal computers or other general purpose applications.

## A microcontroller is a compact microcomputer designed to govern the operation of embedded systems in motor vehicles, robots, office machines, automated sun trackers, and various other devices. A typical microcontroller includes a processor, memory, and peripherals. Arduino UNO R3 is a microcontroller board based on theATmega328 will be used. It has 14 digital input/output pins (of which 6 can be used as Pulse Width Modulation [PWM] outputs), 6 analog inputs, a 16 MHz crystal oscillator, a USB connection, a power jack, an In-Circuit Serial Programming (ICSP)header, and a reset button. The UNO differs from all preceding boards because it does not use the FTDI USB-to-serial driver chip.

## Arduino UNO - JavaTpoint

## Fig.4.1 ARDUINO UNO

## 

## SERVO MOTOR

## A servomotor is a rotary actuator that allows for precise control of angular position. It uses a feedback mechanism to maintain its shaft position with minimum error. The servo motor has three wires. Among those 3 wires, 2 wires provide power supply to the motor and one is used to send signals to the motor for required shaft positions. These signals are PWM signal of varying duty cycle. Servo shaft is rotated by an angle depending on the ON-Time and off time of the pulse. Servo motors are used to rotate the solar panel by sending an electrical pulse of variable width, or pulse width modulation (PWM) through the control wire.

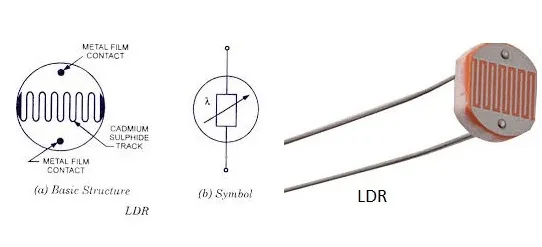
## C:\Users\H\Desktop\71kaoAwYZFL._SL1500_.jpg

## Figure 4.2: Servo Motor

## 3 . LDR [ Light-Dependent Resistor ] & RESISTOR:

LDRs are made of semiconductor materials that exhibit a variation in resistance when exposed to light. When light falls on the LDR, the photons in the light excite the electrons in the semiconductor material, causing a decrease in resistance. The resistance of an LDR can vary significantly based on the intensity of the incident light.

The resistance can drop to a very low value, while in darkness, the resistance can be relatively high. LDR are commonly used in various applications that involve light sensing.



**FIGURE 4.3: LDR**

**4. RESISTOR**

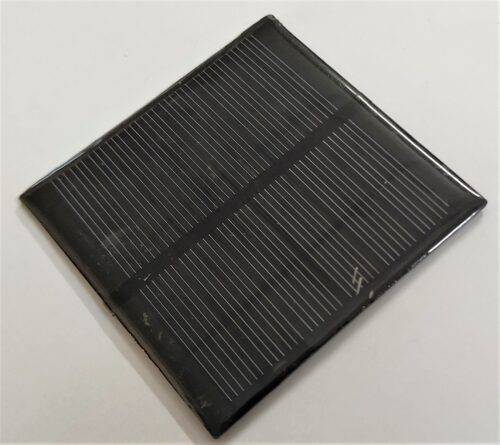
Resistors are commonly used in voltage divider circuits in conjunction with sensors or other components. For example, in a light sensor circuit, a resistor is connected in series with the sensor (e.g., a light-dependent resistor or LDR). The voltage across the resistor is measured to determine the sensor's resistance or provide an analog voltage input to the Arduino.



**FIGURE 4.3 : RESISTOR**

## 5. SOLAR PANEL

The solar panels are securely mounted on a frame or structure that allows them to tilt and rotate along a single axis. This axis is typically aligned in the north-south direction to track the sun's movement from east to west.



**FIGURE 4.4 : SOLAR PANEL**

## SOFTWARE REQUIREMENTS

Software requirements deal with defining resource requirements and prerequisites that need to be installed on a computer to provide the functioning of an application. The minimal software requirements are as follows,

* + 1. Programming Language: C++
    2. Application: Arduino
    3. Operating System: (Windows, Mac, or Linux)
    4. Algorithm: SSD Mode

## C++:

C++ supports various programming paradigms, including procedural programming, object-oriented programming (OOP), and generic programming. It provides a rich set of features such as classes, objects, inheritance, polymorphism, templates, exception handling, and more.

C++ is widely used in various domains, including system programming, game development, embedded systems, high-performance computing, and large-scale applications. Many popular software systems, such as operating systems, browsers, databases, and productivity tools, are built using C++.

1. **Arduino:**

The Arduino Integrated Development Environment (IDE) is the official

software used to write, compile, and upload code to Arduino boards. It is available for free and supports Windows, Mac, and Linux operating systems. You can download the Arduino IDE from the official Arduino website. The Arduino Integrated Development Environment (IDE) is the official software used to write, compile, and upload code to Arduino boards.

It is available for free and supports Windows, Mac, and Linux operating systems. You can download the Arduino IDE from the official Arduino website.

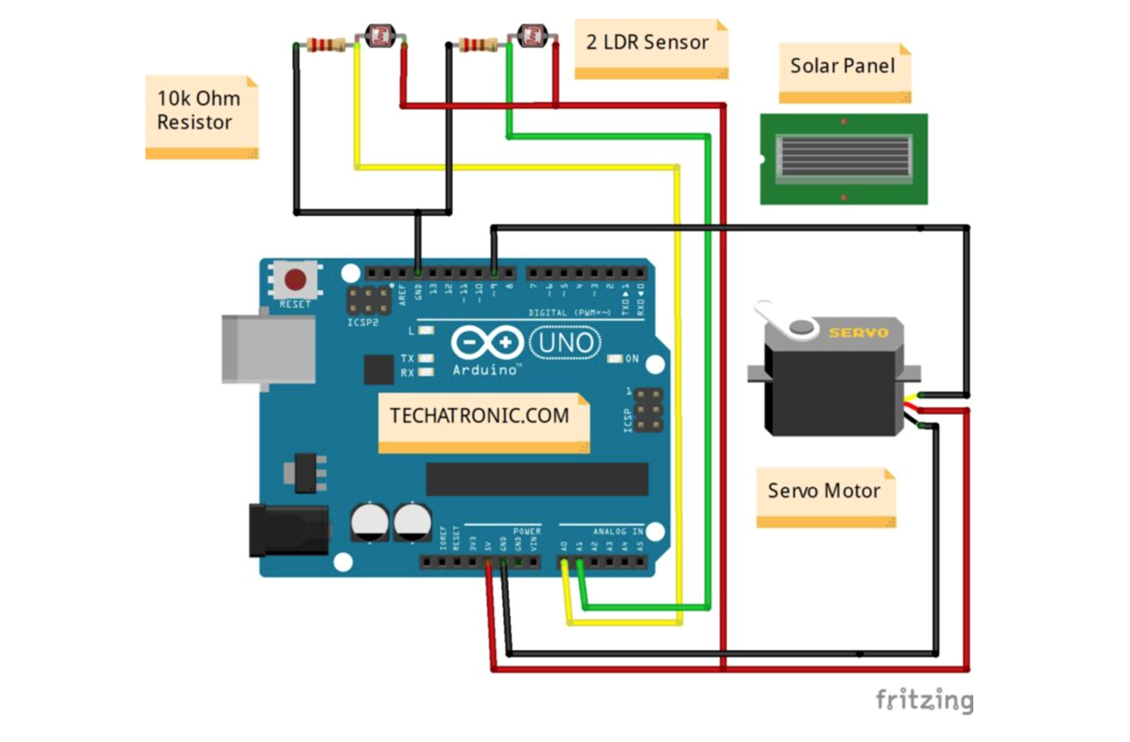
Text Editor or Integrated Development Environment (IDE): While the Arduino IDE is the primary software for Arduino development, you might find it helpful to use a separate text editor or IDE for writing and editing code. Popular options include Visual Studio Code, Sublime Text, Atom, or any other text editor of your choice. These editors often offer advanced features such as code highlighting, code completion, and version control integration.

1. **Operating system:**

Operating System and Drivers: Ensure that you have a compatible operating system (Windows, Mac, or Linux) installed on your computer. Additionally, you may need to install the necessary USB drivers for your specific Arduino board to establish communication between the board and your computer.

## CHAPTER 5 DESIGN ENGINEERING

* 1. **SYSTEM ARCHITECTURE DIAGRAM**

****

## Fig. 5.1 System Architecture Diagram

An architecture diagram is a visual representation of a system's components and their relationships, depicted using shapes and lines. It provides a high-level overview of the system's architecture, including external systems or resources, security controls, and performance metrics. The diagram's specific format and details vary based on the system's complexity and requirements.

## DATA FLOW DIAGRAM

## 

**Fig. 5.2.2 Data Flow Diagram**

A data flow diagram (DFD) is a graphical representation of the "flow" of data through an information system, modeling its process aspects. A DFD is often used as a preliminary step to create an overview of the system without going into detail, which can later be elaborated. DFDs can also be used for the visualization of data processing (structured design).

## USE CASE DIAGRAM

****

**Fig. 5.3 Use Case Diagram**

Use case diagrams overview the usage requirement for system. They are useful for presentations to management and project stakeholders, but for actual development use cases provide significantly more value because they describe the meant of the actual requirements. A use case describes a sequence of action that provides something of measurable value to an action.

## CHAPTER 6 MODULES DESCRIPTION

## IMPORTING NECESSARY LIBRARIES

A library is a collection of functions that can be added to your C++ code and called as necessary, just like any other function. There is no reason to rewrite code that will perform a standard task. With libraries, you can import pre-existing functions and efficiently expand the functionality of your code.

## DATA PREPROCESSING

Analyzing data sets to summarize and visualize data properties is the main area of business. Before considering any inferences, listen to the data by examining all variables in the data.

This EDA process specifically focuses on the following Aspects;

* + 1. Understanding characters of data
    2. Finding meaningful patterns in data
    3. Post modeling strategies
    4. Debugging strategies
    5. Visualization of results

## MODEL DEVELOPMENT

## Data Collection: Gather data related to sunlight intensity, solar panel position, and environmental conditions such as temperature, humidity, and cloud cover. This data will be used to train and optimize the tracking algorithm.

## Algorithm Design: Develop an algorithm that determines the optimal position of the solar panel based on the input data. Consider factors such as sun position, shading, and environmental conditions. The algorithm should be designed to maximize energy generation and minimize tracking errors.

## Iterative Improvement: Analyze the performance of the algorithm and identify

## areas for improvement. Iterate on the algorithm design, making adjustments and

## fine-tuning to enhance its tracking capabilities.

## MODEL EVALUATION

Performance metrics to evaluate the effectiveness of the solar tracker system. Metrics may include energy generation efficiency, tracking accuracy, and response time to changing conditions. Conduct extensive testing of the solar tracker system under various scenarios and conditions. Measure the system's performance against the defined metrics to assess its effectiveness .Compare the performance of the solar tracker system with a fixed solar panel configuration. Analyze the energy generation difference and demonstrate the benefits of using the tracking system.

## MODEL DEPLOYMENT

## Integrate the Arduino board, light sensor, motor, and other necessary components into a physical setup. Ensure proper wiring and connections between the components. Program the Arduino board with the finalized algorithm and logic for controlling the motor and tracking the sun. Ensure the software is properly uploaded to the Arduino board. Set up a monitoring system to track the solar panel's performance and gather data on energy generation. Implement a maintenance plan to regularly inspect and clean the system, check for any hardware issues, and make necessary adjustments.

## INTRODUCTION

**CHAPTER 7 SOLAR TECHNOLOGY**

Solar tracking technology is a technique used to optimize the positioning of solar panels or solar collectors to maximize their exposure to sunlight throughout the day. By tracking the sun's movement across the sky, solar tracking systems ensure that solar panels are always oriented at an optimal angle relative to the sun, resulting in increased energy generation and improved efficiency of solar power systems.

The main objective of solar tracking technology is to capture as much solar energy as possible by maintaining a perpendicular angle between the solar panels and the incoming sunlight. This allows the panels to receive direct sunlight for a longer duration, enhancing the overall energy output.

Solar tracking technology offers several advantages over fixed solar panel configurations. By following the sun's path, solar tracking systems can significantly increase energy generation, typically resulting in 20% to 40% more output compared to fixed installations. This higher energy yield makes solar tracking systems particularly beneficial in regions with varying sunlight angles or during seasons with shorter daylight hours.

Implementing solar tracking technology can contribute to the overall feasibility and economics of solar power systems. While solar tracking systems require additional components, such as motors, sensors, and control systems, the increased energy production can offset the initial investment and lead to a faster return on investment.

## WORKING PRINCIPLE

## Light Sensor Detection: The solar tracker system incorporates a light sensor, such as a Light Dependent Resistor (LDR), to detect the intensity of sunlight. The LDR measures the amount of light falling on it and provides analog voltage readings to the Arduino board.

## Sun Position Calculation: Using the analog voltage readings from the LDR, the Arduino board calculates the position of the sun relative to the solar panel. This can be done by mapping the analog readings to corresponding sun positions or by using mathematical calculations based on the known characteristics of the LDR and the sun's path.

## Motor Control: The Arduino board controls a motor that adjusts the orientation of the solar panel along the single-axis (typically east-west). The motor can be a DC motor or a servo motor, depending on the design of the solar tracker system.

## Tracking Algorithm: The Arduino board runs a tracking algorithm that determines the desired position of the solar panel based on the calculated sun position. The algorithm considers factors such as the time of day, the season, and the desired tilt angle for maximum solar energy capture.

## Motor Movement: Based on the desired position calculated by the tracking algorithm, the Arduino sends control signals to the motor to move the solar panel accordingly. The motor rotates the panel to align it with the optimal angle for sun exposure.

## Continuous Monitoring and Adjustment: The Arduino continuously monitors the sunlight intensity using the LDR and recalculates the sun's position. It compares the current position of the solar panel with the desired position and makes necessary adjustments to ensure accurate tracking.

## COMPONENTS

The basic components of a drone are the frame, motors, propellers, battery, flight controller, and sensors. Let us take a closer look at each of these components.

* Arduino Uno
* Servo motor
* Solar panel
* LDR
* Resistor(10k ohm)

## Classification of Solar Tracker;

● Single-Axis Trackers

● Dual-Axis Trackers

● Azimuth-Altitude Trackers

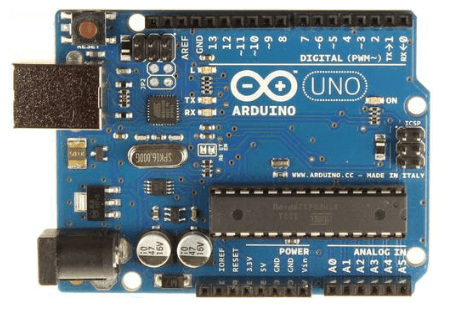
● Passive Trackers

● Time-Based Trackers

● Active Trackers

## Arduino

The Arduino programming language is used to program microcontroller boards such as the Arduino Uno to interact with sensors, actuators, and other devices connected to the board. In fact, the language is based on C++, and it is designed to be easy to use for beginners and non-programmers.



## Fig. 7.1 Arduino Board

1. **Servo motor**

A brushless DC (BLDC) motor is a type of electric motor that relies on repulsive and attractive forces between permanent magnets and electromagnets to drive its

rotation. Due to the lack of friction, BLDC motors require less maintenance and have a longer lifespan.

They are more efficient than brushed motors and have a higher torque to weight ratio. Many DC motors use brushes to transfer current from the commutator to the rotor.



## Fig. 7.2 Servo Motor MG90s

1. **LDR**

Photoresistors, also known as light dependent resistors (LDR), are light sensitive devices most often used to indicate the presence or absence of light, or to measure the light intensity



## Fig. 7.3 LDR

1. **Solar panel**

The solar panels are securely mounted on a frame or structure that allows them to tilt and rotate along a single axis. This axis is typically aligned in the north-south direction to track the sun's movement from east to west.



**Fig. 7.4 Solar panel**

1. **Resistor & Wire**

Resistor is used to resistance the flow of current. When resistor is placed in a circuit, the current flow decreases when current passes through the resistor. It is used to interconnect the components of a breadboard or other prototype or test circuit internally or with other equipment or components without soldering.



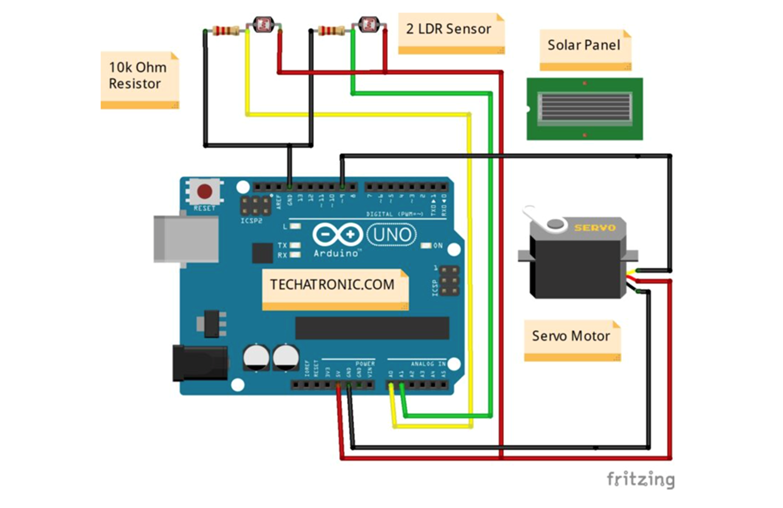
## Fig. 7.5 Resistor

## Buy MALE TO MALE JUMPER WIRE : ElementzOnline INDIA

## Fig 7.6 Jumper Wire

## CHAPTER 8 IMPLEMENTATION

**8.1 : CONNET THE HARDWARECOMPONENTS:**



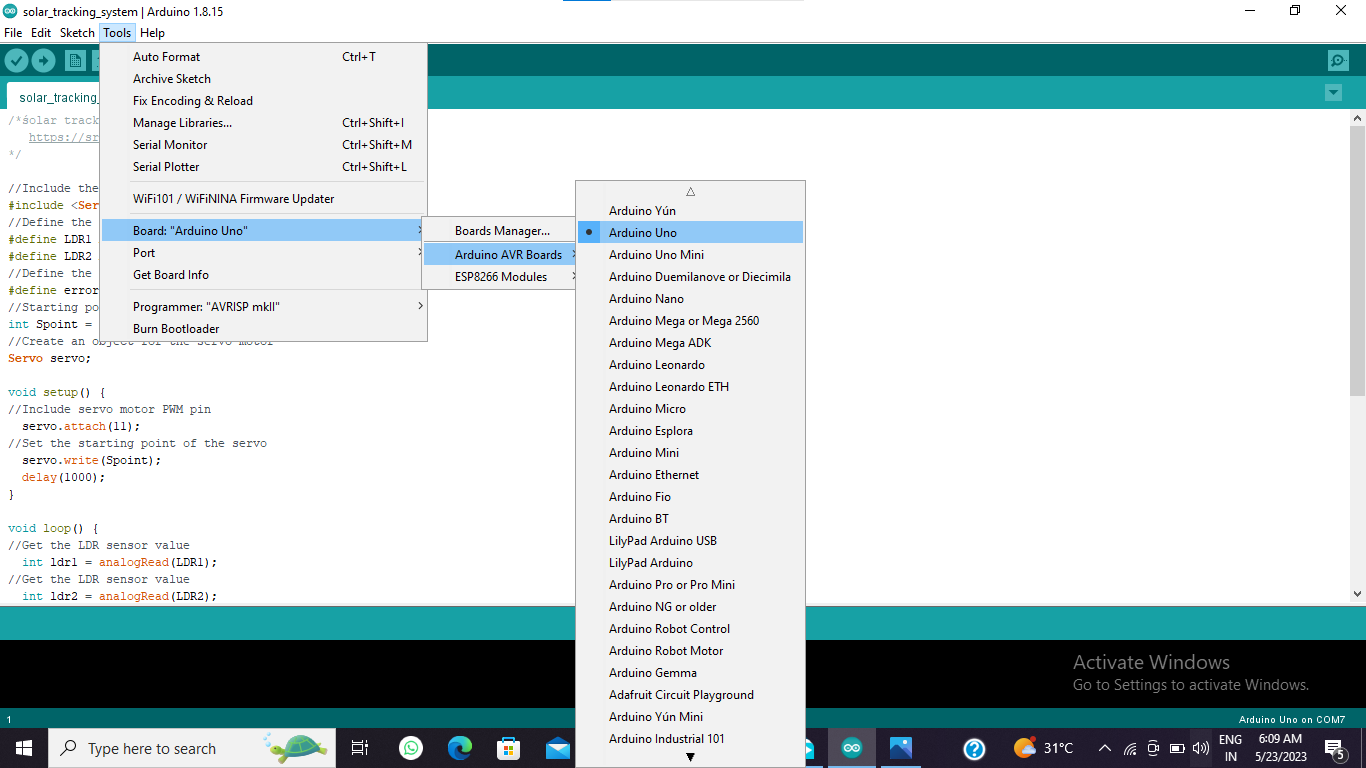
Using this architecture

**8.2 : DOWNLOAD & INSTALL ARDUINO APPLICATION:**

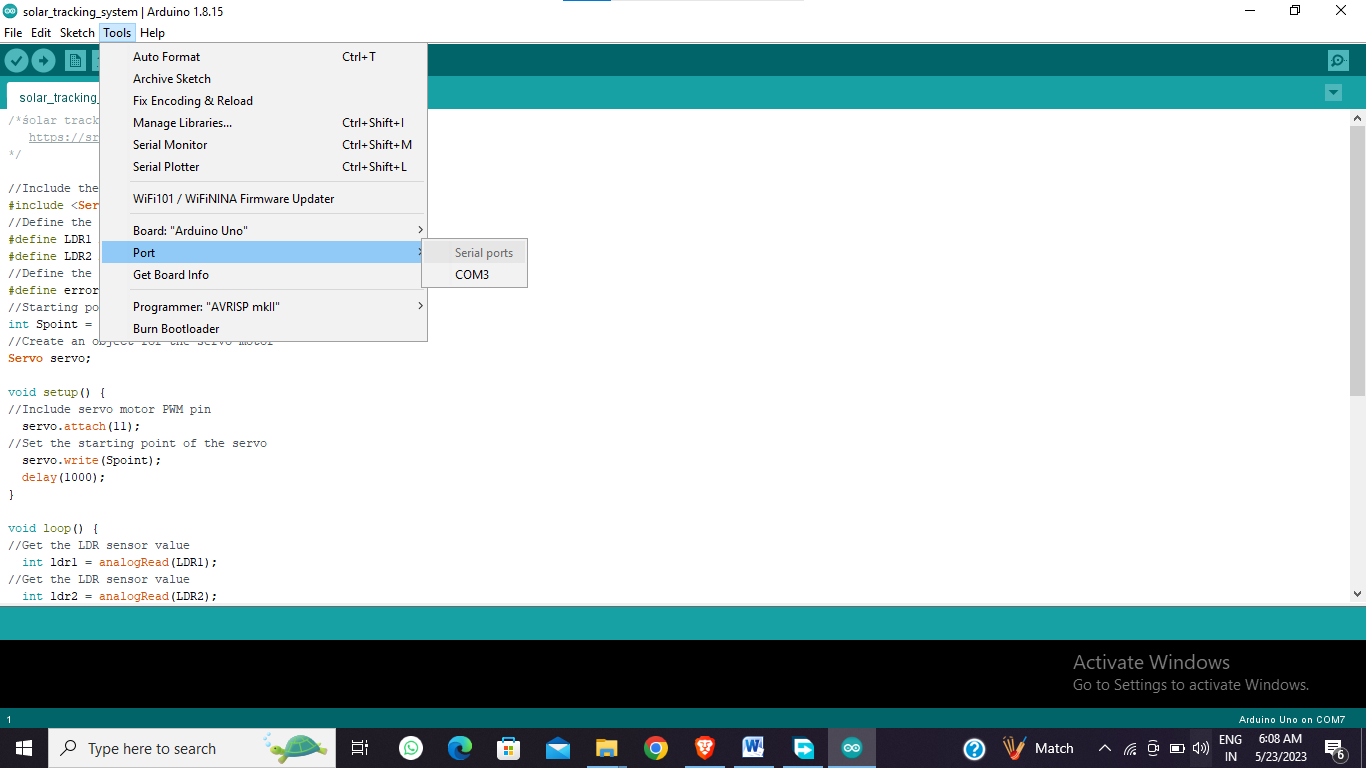
https://www.arduino.cc/en/donate/

**8. 3: ARDUINO SETUP:**

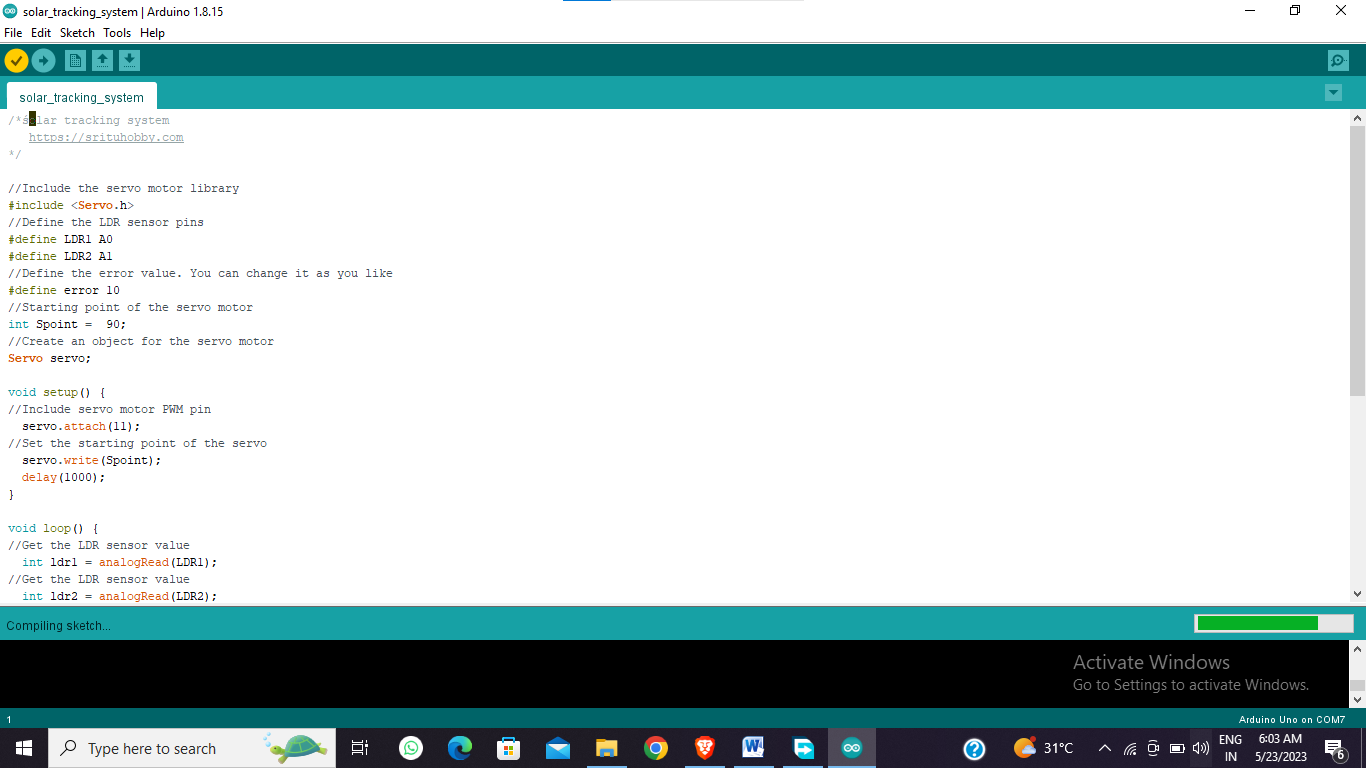
* open arduino application select tool then goto board in that select arduino AVR board then select arduino uno

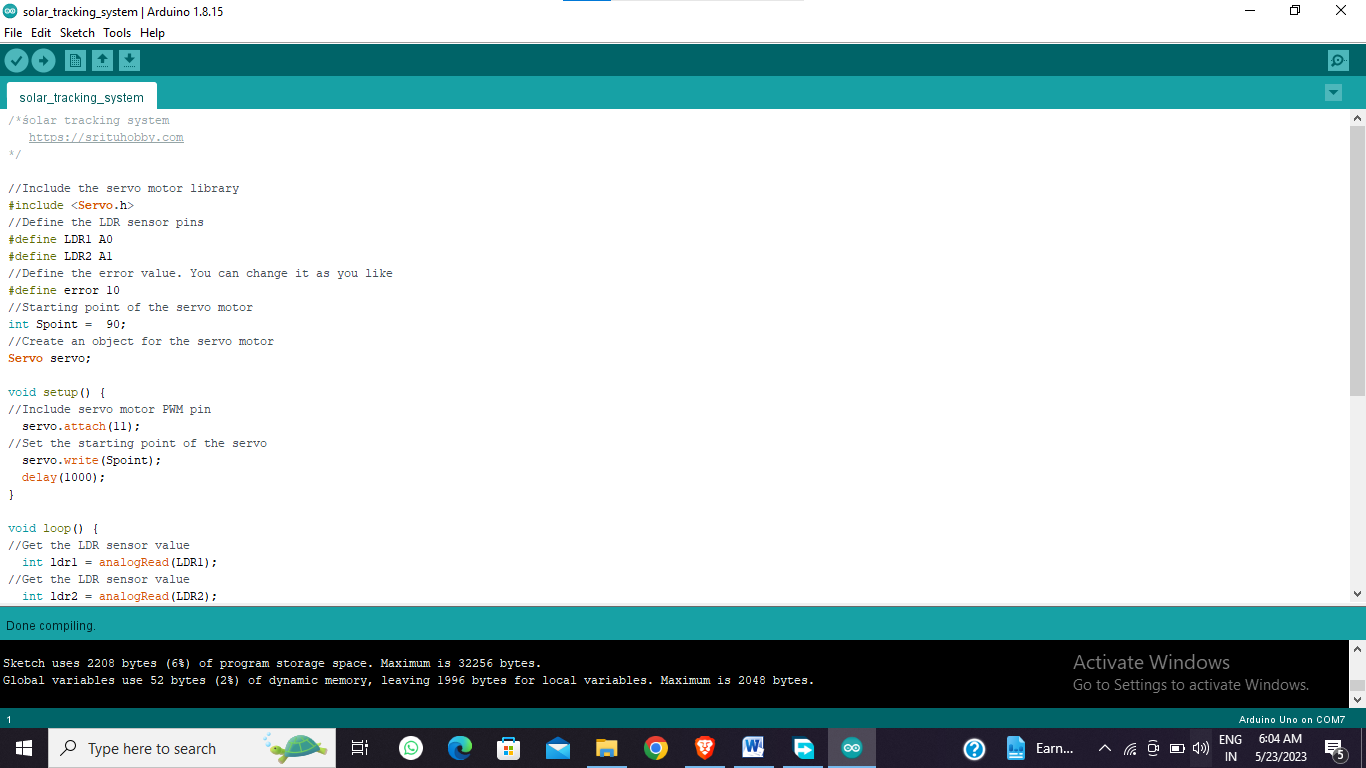


* connect arduino board to system with usb and then select the port

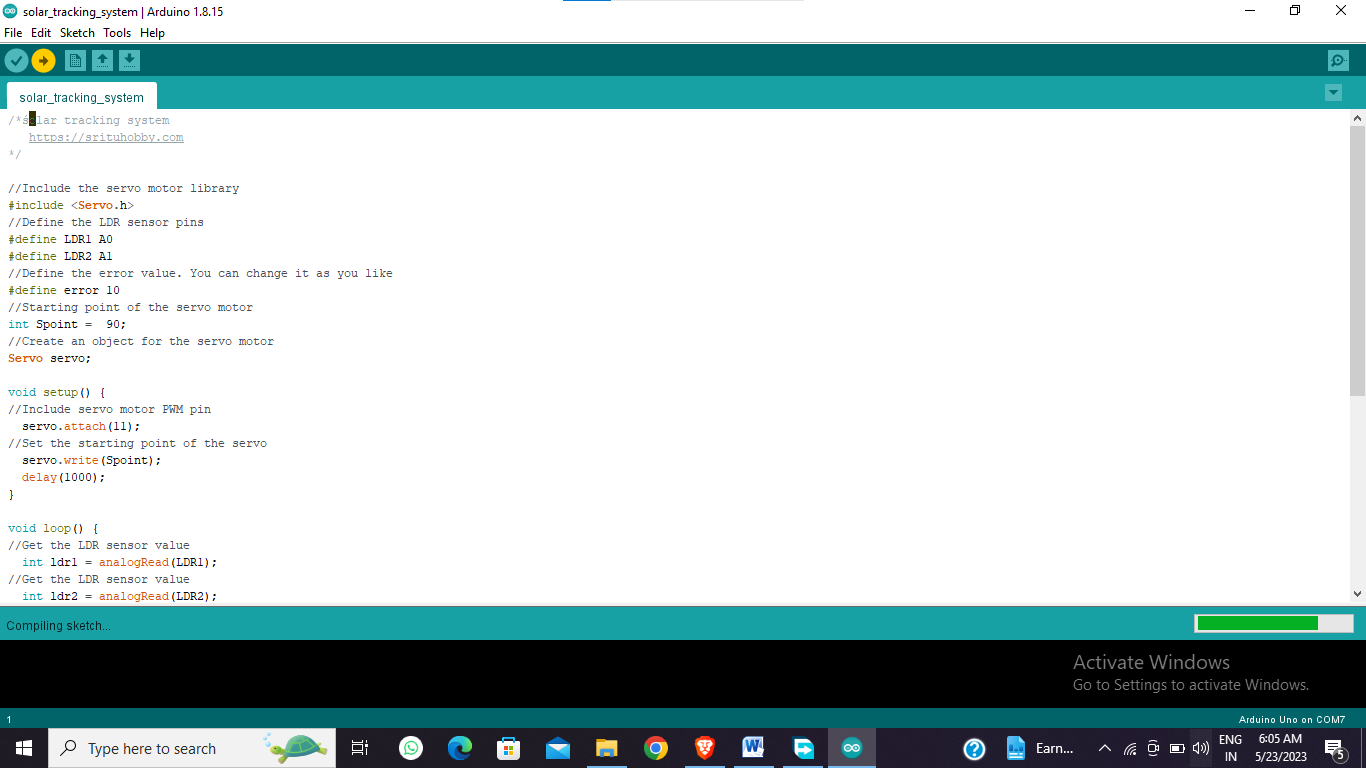


* Verify the program





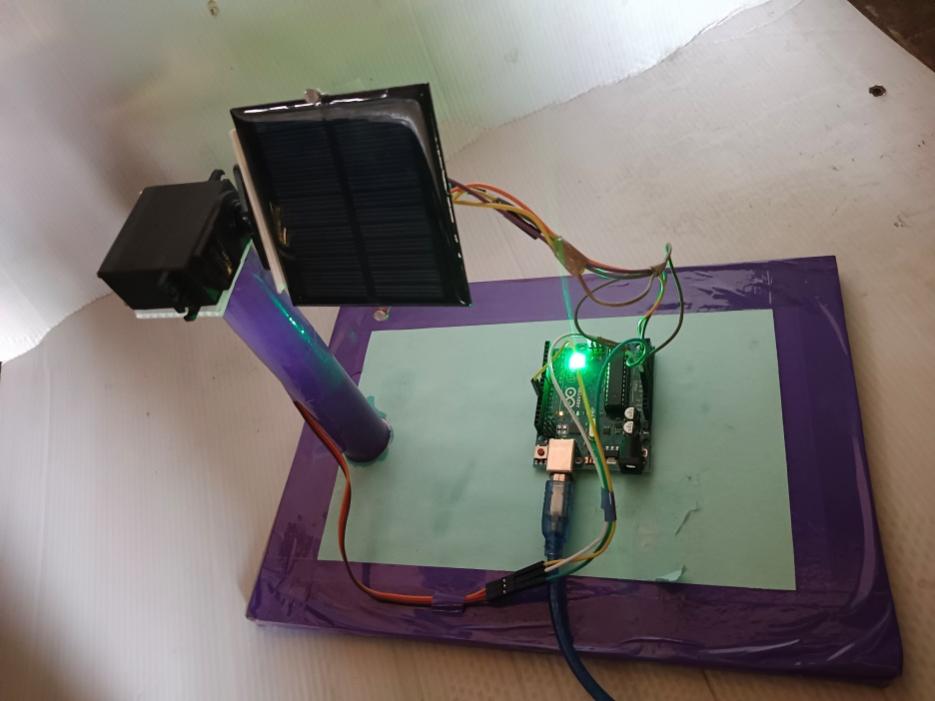
* Upload the program into arduino uno



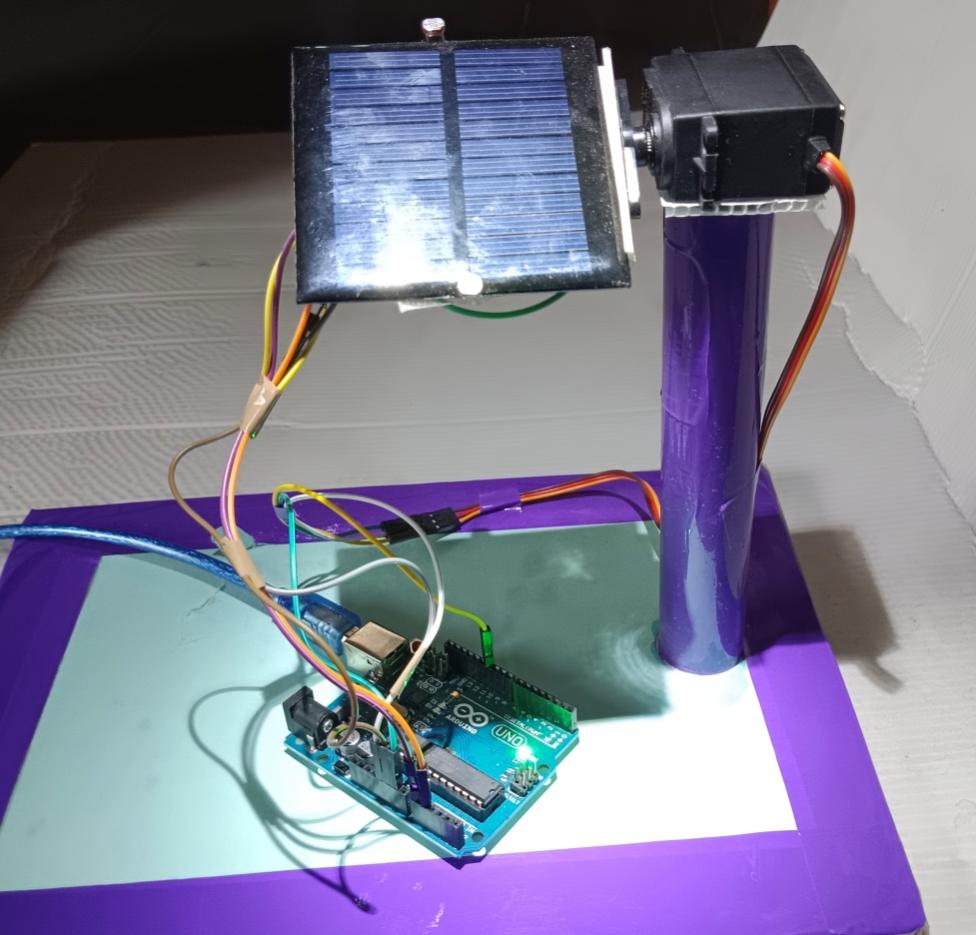
**CHAPTER 9**

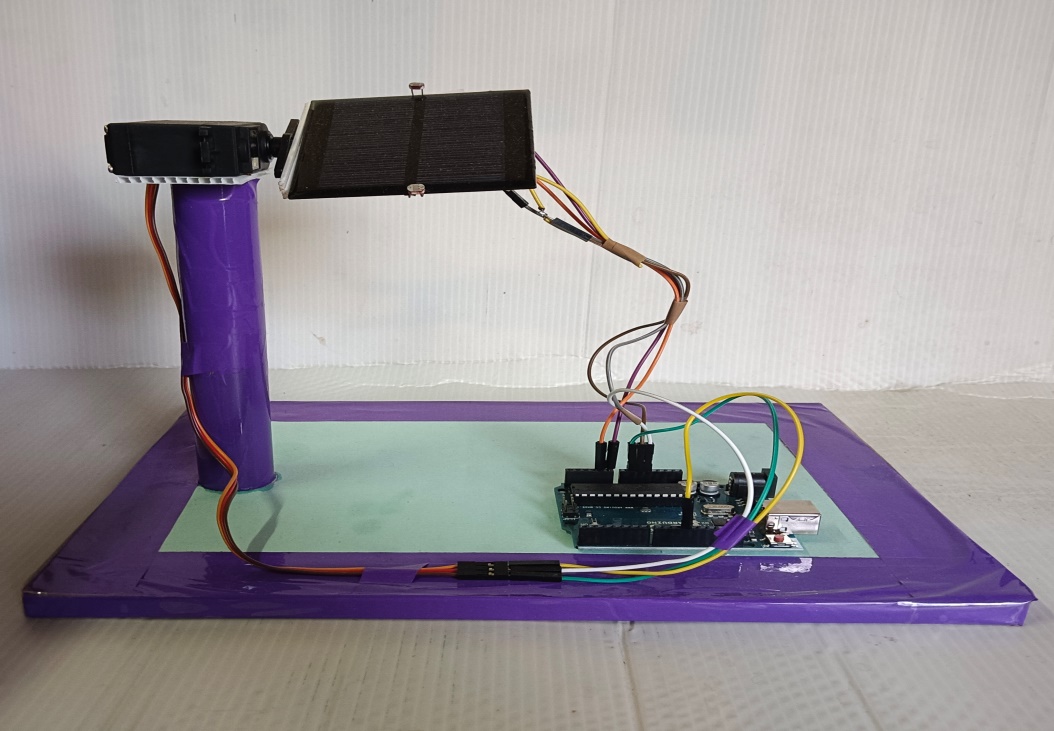
**SNAPSHOTS**

**9.1 SAMPLE INPUT**

****

**9.2 SAMPLE OUTPUT**





**PROGRAM**

**// Techatronic.com**

#include <Servo.h>  **//including the library of servo motor**

Servo sg90;

int initial\_position = 90;

int LDR1 = A0; **//Connect The LDR1 on Pin A0**

int LDR2 = A1; **//Connect The LDR2 on pin A1**

int error = 5;

int servopin=9; **//You can change servo just makesure its on arduino's PWM** pin

void setup()

{

sg90.attach(servopin);

pinMode(LDR1, INPUT);

pinMode(LDR2, INPUT);

sg90.write(initial\_position); **//Move servo at 90 degree**

delay(2000);

}

void loop()

{

int R1 = analogRead(LDR1); // **read LDR 1**

int R2 = analogRead(LDR2); **// read LDR 2**

int diff1= abs(R1 - R2);

int diff2= abs(R2 - R1);

if((diff1 <= error) || (diff2 <= error)) {

} else {

if(R1 > R2)

{

initial\_position = --initial\_position;

}

if(R1 < R2)

{

initial\_position = ++initial\_position;

}

}

sg90.write(initial\_position);

delay(100);

}

## CHAPTER 10

**CONCLUSION AND FUTURE ENHANCEMENT**

## CONCLUSION

In conclusion, a single-axis solar tracker using Arduino is a valuable technology that enhances the efficiency and performance of solar panel systems. By continuously adjusting the position of the solar panel to align with the sun's movement, the tracker maximizes sunlight exposure and improves energy generation.Throughout this project, we have explored the existing system of solar trackers, the working principle of a single-axis tracker using Arduino, and various components involved in its implementation. We have also discussed the importance of data preprocessing, algorithm selection, and evaluation metrics in developing an effective solar tracking system.

* 1. **FUTURE ENHANCEMENT**

In terms of future enhancements, can further improve the system's functionality, such as integrating additional sensors, advanced tracking algorithms, wireless communication, and data analytics. These enhancements will contribute to increased energy efficiency, better performance monitoring, and remote control capabilities .Overall, a single-axis solar tracker using Arduino offers a promising approach to harnessing solar energy more effectively. It reduces dependency on manual adjustments, maximizes energy generation, and ultimately contributes to a more sustainable and renewable energy future.

## REFERENCES

* + 1. A. Sharma, A. Srivastava, and N. Arora, "Design and Performance Evaluation of a Single-Axis Solar Tracker for Maximum Energy Yield" pp. 531-548, 2017.
    2. M. S. Ahmed, M. R. Karim, and M. H. Ali, " "Design and Development of Low-Cost Single Axis Solar Tracking System"," International Conference on Electrical Engineering and Information & Communication Technology (ICEEICT), pp. 1-6, 2017.
    3. K. Chinh, T. A. Nguyen, and H. T. Duong, " "Development and Analysis of a Low-Cost Single-Axis Solar Tracking System for Concentrated Photovoltaic Applications", pp. 1-6, 2016.
    4. R. K. Raut, A. B. Kadam, and K. K. Kathane, " "Performance Analysis of a Single-Axis Sun Tracking System for Solar Panels, " International Conference on Electrical, Electronics, and Optimization Techniques (ICEEOT), pp. 2510-2515, 2016.
    5. B. Patil, N. Gupta, and P. Singh " "Design and Implementation of a Microcontroller-Based Single Axis Solar Tracking System"," International Journal of Engineering Research & Technology, pp. 8-12, 2016.
    6. A. D. Thakur and S. S. Wagh ", "Design and Development of an Arduino-Based Single Axis Solar Tracking System”, International Journal of Innovative Research in Science, Engineering and Technology, pp. 10967-10972, 2015.
    7. S. Rathinasamy and R. Hemapriya", "A Study on Single Axis Solar Tracking System and Its Power Optimization"," International Conference on Green Computing and Internet of Things (ICGCIoT), pp. 836-840, 2015.
    8. S. Kumar, S. Kumar, and S. M. Thakur, " "Design and Development of Single Axis Sun Tracking System Using Microcontroller International Conference on Advances in Electronics, Computers and Communications (ICAECC), pp. 43-47, 2014.
    9. M. M. Rahman, M. M. R. Bhuiya, and M. A. R. Sarkar, " "Design and Implementation of a Microcontroller-Based Single-Axis Solar Tracking ystem," International Journal of Electrical and Computer Engineering Volume: 3, pp. 603-610, 2013.
    10. R. Gupta, P. Kumar, and S. C. Kaushik, "Performance Analysis of Single-Axis Sun Tracking System for Maximum Solar Energy"," International Journal of Computer Applications, pp 1-5, 2013.
    11. R. Gupta, P. Kumar, and S. C. Kaushik, "Performance Analysis of Single-Axis Sun Tracking System for Maximum Solar Energy"," International Journal of Computer Applications, pp 1-5, 2013.
    12. M. A. Rahman, M. Hasan, and A. Iqbal, "A Review of Single Axis Sun Tracking Techniques for Efficient Solar Photovoltaic System, International Conference on Informatics, Electronics & Vision (ICIEV), pp. 679-684, 2012.