Dual Axis Solar Tracker

1[st] Md.Shahriar Rahman Anuvab
21101094

2[nd] Mishkat Sultana
19201028

3[rd]Nuzhat Tabassum
19101537

4[th]Nuzhat Chowdhury
19101262

***Abstract*—This project showcases a self-adjusting system that tracks the strongest light source using servo motors and light-dependent resistors (LDRs). By strategically positioning LDRs and utilizing servos, the system detects and follows light sources accurately. Arduino code processes LDR inputs, calculates average values, and guides servo movements to align with the light. The project demonstrates an innovative use of basic components for autonomous light tracking, holding potential in applications such as solar technology and responsive robotics.**

***Index Terms*—component, formatting, style, styling, insert**

I. Introduction

Renewable energy sources have attracted the attention of researchers, technologists, investors, and decision makers worldwide due to the inevitable future lack of fossil fuel supplies. Among the new energy sources receiving attention are hydroelectricity, bioenergy, tidal power, wave power, solar, wind, and geothermal energy. They are viewed as advantageous substitutes for fossil fuel sources due to their renewability. Among Solar photovoltaic (PV) energy is one of the most readily available sources of these energy forms. Due to research and development efforts to increase the efficiency and decrease the cost of solar cells, this technology is currently being used more frequently in homes. The International Energy Agency (IEA) reports that since the early 2000s, the average annual growth rate for global PV capacity has been 49

Despite the benefits, solar PV energy is still a long way from displacing existing conventional sources. Maximizing the power production of PV systems in places with little solar radiation is still a challenge. Although we still need manufacturers to develop more sophisticated technologies to increase the capabilities of PV materials, improving system design and module building is a practical way to increase solar PV power's efficiency and make it a more reliable option for clients. This project was carried out with that objective in mind to aid in the advancement of such a promising technology.

Maximizing time spent in the sun is one of the key ways to increase effectiveness. By ensuring that PV solar panels are always aligned with the sun's rays at the proper angle, tracking systems assist in achieving this. The purpose of this project is to construct a scaled-down prototype of a light tracking system; however, the design can be used for any solar energy system in real life. A quantitative evaluation of the tracking system's performance in comparison to systems with fixed mounting techniques is also anticipated from this study.

II. Working Principle

- Light-dependent resistance (LDR) is dependent on light intensity and fluctuates as a result. When the light intensity is low, the LDR resistance is higher and a higher output voltage is obtained. Conversely, when the light intensity is high, the LDR resistance is lower and a lower output voltage is obtained.
- In order to obtain the output voltage from the sensors (LDRs), a potential divider circuit is utilized. This image displays the circuit.
- The LDR detects analog input voltages of 0 to 5 volts and outputs a digital number that typically falls between 0 and 1023.
- This will now use the Arduino software (IDE) to provide feedback to the microcontroller.
- This mechanism, which is covered later in the hardware model, is capable of controlling the position of a servo motor.
- After determining the maximum intensity of light falling perpendicular to it, the tracker finally modifies its position and maintains it until it detects any additional changes.
- The point source of light affects the LDR's sensitivity. It rarely has an impact on the state of diffuse lighting.

III. BASIC CIRCUIT DIAGRAM

An overview of the required circuit for the Dual-axes solar tracker is shown here.
The 5V supply is fed from an USB 5V dc voltage source through Arduino Board.
Servo X : Rotates solar panel along X direction
Servo Y : Rotates solar panel along Y direction

IV. Mathmatical Model

A. Inverse Square Law

A surface's illumination changes inversely as the square of its distance from the source. As a result, if a surface is illuminated at a distance of I units from the source, the illumination at 2 meters will be I/4, at 3 meters will be I/9, and so on. In fact, inverse square law operates only when the light rays are from a point source and are incident normally upon the surface. Thus illumination in lamberts/m2 on a normal plane= Candle power/ (Distance in metres)$\hat{2}$

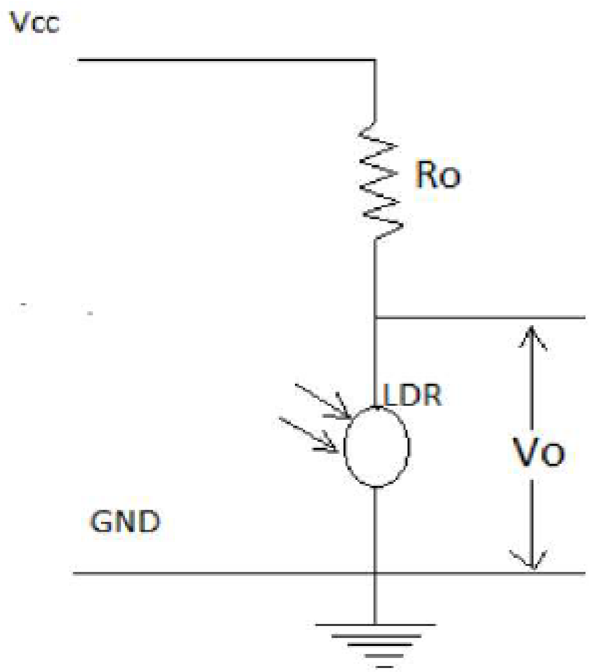

Fig. 1. Example of a figure caption.

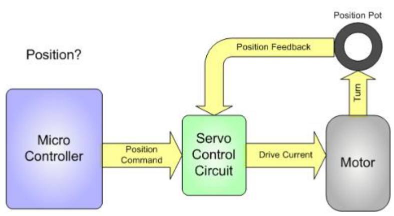

Fig. 2. Example of a figure caption.

B. LAMBERT'S COSINE LAW

The illumination received on a surface is proportional to the cosine of the angle between the direction of the incident light rays and normal to the surface at the point of incidence. This is mainly due to the reduction of the projected area as the angle of incidence increases.

The equations are:

$$E_\theta = E\cos\theta = I\cos\theta \cdot D^2 \tag{1}$$

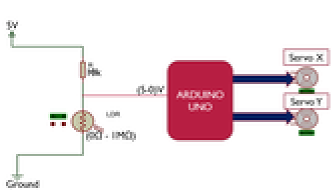

Fig. 3. Example of a figure caption.

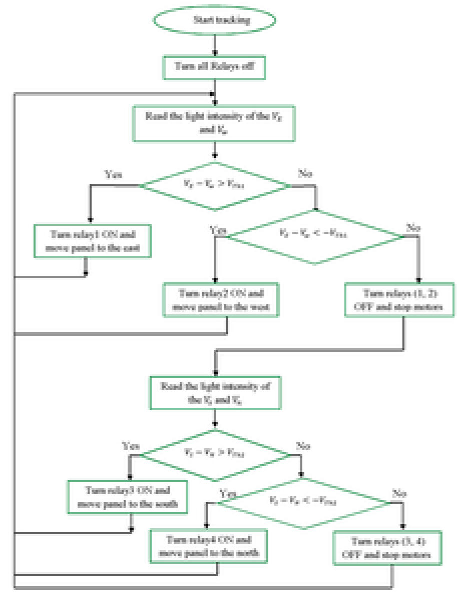

Fig. 4. Example of a figure caption.

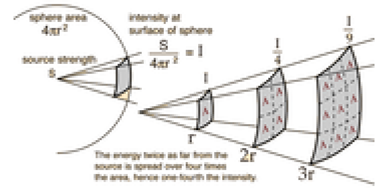

Fig. 5. Example of a figure caption.

where,

E_θ = illumination on horizontal plane
E = illumination due to light normally incident
θ = the angle of incidence
D = distance from the surface

V. HARDWARE MODEL

A. EXPLANATION OF THE BLOCK DIAGRAM:

Three Light Dependent Resistors (LDRs), as shown in the block diagram, are arranged side by side on a plate with a solar panel. Different amounts of light from a source are reflected off of them. All LDRs' resistance values are not consistently the same because of their inherent photoconductivity, which causes resistance to decrease as incident light intensity increases.

Each LDR transmits an equivalent signal representing the value of its corresponding resistance to the microcontroller, which is set up using the necessary programming logic. The values are contrasted with one another using a certain LDR value as a benchmark.

The driving axle of one of the two DC servo motors is mechanically linked to the other, allowing the latter to rotate while the former moves. A solar panel is driven by the former servo motor's axle. The two servo motors are set up so that the solar panel may move both along the X-axis and the Y-axis.

Based on the input signals received from the LDRs, the microcontroller provides the proper impulses to the servo motors. Using one servo motor for x-axis tracking and the other for y-axis tracking.

In this way the solar tracking system is designed.

B. *ARDUINO UNO*

A microcontroller board called the Arduino Uno is based on the ATmega328. The open-source prototyping platform Arduino is perfect for both professionals and enthusiasts to use because of its simplicity. The Arduino Uno contains a 16 MHz crystal oscillator, 6 analog inputs, 14 digital input/output pins (of which 6 can be used as PWM outputs), a USB port, a power jack, an ICSP header, and a reset button. It comes with everything required to support the microcontroller; to get started, just use a USB cable to connect it to a computer, or an AC-to-DC adapter or battery to power it. The FTDI USB-to-serial driver chip is not used by the Arduino Uno, setting it apart from all previous boards. Instead, it has an Atmega8U2 microcontroller chip that has been configured to function as a USB to serial converter.

Italian for "one," "Uno" was chosen to symbolize the imminent Arduino 1.0 release. Moving forward, the Arduino Uno and version 1.0 will serve as the standard versions of Arduno. The Uno, the most recent in a line of USB Arduino boards, serves as the platform's benchmark.

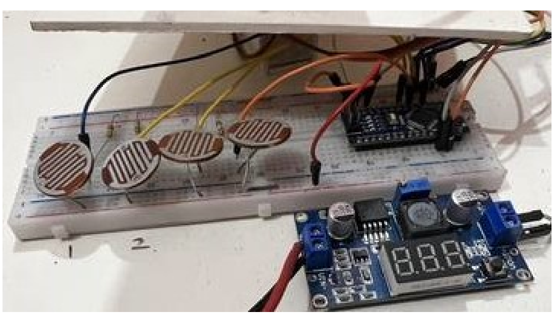

Fig. 6. Example of a figure caption.

C. *ADC CONCEPT IN ARDUINO UNO*

Arduino uno board has 6 ADC input ports. Among those any one or all of them can be used as inputs for analog voltage. The Arduino Uno ADC is of 10 bit resolution (so the integer values from (0-(2Î0) 1023)). This means that it will map input voltages between 0 and 5 volts into integer values between 0 and 1023. So, for every (5/1024= 4.9mV) per unit. The UNO ADC channels have a default reference value of 5V. This means we can give a maximum input voltage of 5V for ADC conversion at any input channel. Since some sensors provide voltages from 0-2.5V, with a 5V reference we get lesser accuracy, so we have a instruction that enables us to change this reference value. So for changing the reference value we have. As default we get the maximum board ADC resolution which is 10bits, this resolution can be changed by using instruction .

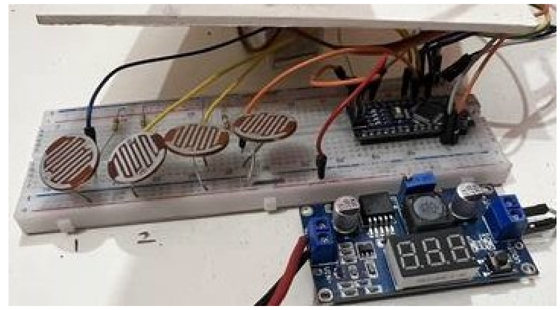

Fig. 7. Example of a figure caption.

VI. SOFTWARE PROGRAM MODEL

A. *PROGRAMMING CODE:*

```
include <Servo.h>

Servo servohori;
int servoh = 0;
int servohLimitHigh = 180;
int servohLimitLow = 10;

Servo servoverti;
int servov = 0;
int servovLimitHigh = 180;
int servovLimitLow = 10;

int ldrtopr = 1;
int ldrtopl = 2;

int ldrbotr = 0;
int ldrbotl = 3;


void setup ()

servohori.attach(10);
servohori.write(0);

servoverti.attach(9);
servoverti.write(0);
delay(500);


void loop()

servoh = servohori.read();
servov = servoverti.read();

int topl = analogRead(ldrtopl);
int topr = analogRead(ldrtopr);
int botl = analogRead(ldrbotl);
```

int botr = analogRead(ldrbotr);

int avgtop = (topl + topr) / 2;
int avgbot = (botl + botr) / 2;
int avgleft = (topl + botl) / 2;
int avgright = (topr + botr) / 2;

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);

delay(50);

VII. DESCRIPTION OF THE SOFTWARE PROGRAM

A. Steps :

- First of all, both the servos are declared and object is created to control the servo motors.
- The variables posx and posy are used to store the reference servo positions.
- The ADC input pins for LDRs are selected for dual direction movement and one for reference.
- A tolerance or a constant value is selected to establish the working of the motors.
- The servos are attached on digital pins to the servo object.
- The required analog pins are selected as input using pinMode(pin , mode)
- The servos are sets to mid-point or original position with a 1000ms or 1sec delay to catch up with the user.
- Three variables are chosen to read the analog values and map it into integers value between 0 and 1023.
- If the difference between the two variables is less than the tolerance value then it will stays to its or original location else it shows movement towards the direction of maximum intensity of light by incrementing or decrementing the values of posx and posy.
- The position is then written to servo and the loop repeats till it encounter any changes in the values of input greater than the minimum tolerance.
- If the position becomes greater than 150then position will be set to 150only and if the position of the motor is less than 30then it would be kept at 30only as the lower and upper limit angles are chosen to be 30and 150respectively.

Be sure that the symbols in your equation have been defined before or immediately following the equation. Use “(**??**)”, not “Eq. (**??**)” or “equation (**??**)”, except at the beginning of a sentence: “Equation (**??**) is . . .”

VIII. LDR PROGRAM AND GRAPH

void setup() Serial.begin(9600); void loop()
int sensorValue = analogRead(A0); Serial.println(sensorValue); delay(10);

IX. CODE RELATING ANALOG TO DIGITAL CONVERSION

In the program below, the very first thing that you do will in the setup function is to begin serial communications, at 9600 bits of data per second, between your board and your computer with the line:

Serial.begin(9600);

Next, in the main loop of your code, you need to establish a variable to store the resistance value (which will be between 0 and 1023, perfect for an int datatype) coming in from your potentiometer:

int sensorValue = analogRead(A0);

To change the values from 0-1023 to a range that corresponds to the voltage the pin is reading, you'll need to create another variable, a float, and do a little math. To scale the numbers between 0.0 and 5.0, divide 5.0 by 1023.0 and multiply that by sensorValue :

float voltage= sensorValue * (5.0 / 1023.0);

Finally, you need to print this information to your serial window as. You can do this with the command Serial.println() in your last line of code:

Serial.println(voltage)

Now, when you open your Serial Monitor in the Arduino IDE (by clicking on the icon on the right side of the top green bar or pressing Ctrl+Shift+M), you should see a steady stream of numbers ranging from 0.0 - 5.0. As you turn the pot, the values will change, corresponding to the voltage coming into pin A0.

X. ABOUT SOLAR PANEL AND CONNECTED LOAD

Solar panel is placed at the top and connected to a load directly. The load may a led or a voltmeter which could be connected to get the exact voltage which depends on the intensity of light falling on the panel and the position of the tracker. Ø Concentrated solar photovoltaics' and have optics that directly accept sunlight, so solar trackers must be angled correctly to collect energy. All concentrated solar systems have trackers because the systems do not produce energy unless directed correctly toward the sun. Ø The solar panel is just a mere device to accept the light radiation which is purely controlled by LDR sensors and the load connected depends upon the rating of the panel used.

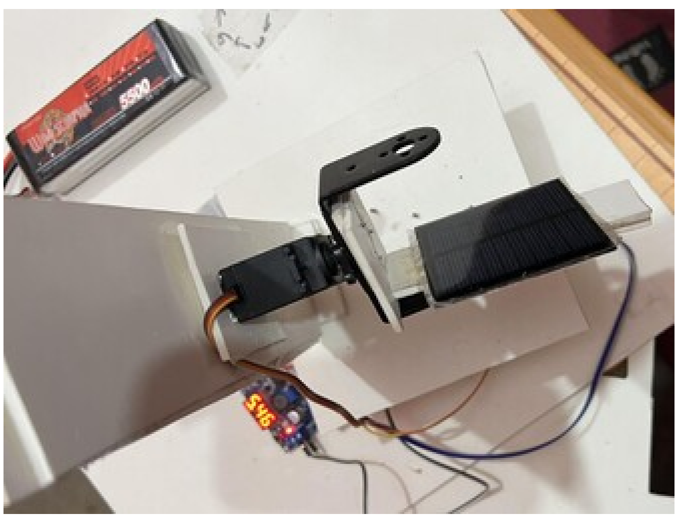

Fig. 8. Example of a figure caption.

XI. DUAL AXIS MOVEMENT OF SOLAR TRACKER

The dual axis solar tracker is device which senses the light and positions towards the maximum intensity of light. It is made in such a way to track the light coming from any direction.
To simulate the general scenario of the Sun's movement, the total coverage of the movement of the tracker is considered as 120 in both the directions.
The initial position of both the servo motors are chosen at 90i.e, for east-west servo motor as well as for north-south servo motor.
The position of the tracker ascends or descends only when the threshold value is above the tolerance limit.

XII. BENEFITS AND DEMERITS OF SOLAR ENERGY

A. *Benefits:*

- Solar energy is a clean and renewable energy source. Once a solar panel is installed, the energy is produced at reduced costs.
- Whereas the reserves of oil of the world are estimated to be depleted in future, solar energy will last forever.
- It is pollution free.
- Solar cells are free of any noise. On the other hand, various machines used for pumping oil or for power generation are noisy.
- Once solar cells have been installed and running, minimal maintenance is required. Some solar panels have no moving parts, making them to last even longer with no maintenance.
- On average, it is possible to have a high return on investment because of the free energy solar panels produce.
- Solar energy can be used in very remote areas where extension of the electricity power grid is costly.

B. *Demerits:*

- Solar panels can be costly to install resulting in a time lag of many years for savings on energy bills to match initial investments.
- Generation of electricity from solar is dependent on the country's exposure to sunlight. That means some countries are slightly disadvantaged.
- Solar power stations do not match the power output of conventional power stations of similar size. Furthermore, they may be expensive to build.
- Solar power is used for charging large batteries so that solar powered devices can be used in the night. The batteries used can be large and heavy, taking up plenty of space and needing frequent replacement.
- As the merits are more than the demerits, the use of solar power is considered as a clean and viable source of energy. The various limitations can be reduced through various ways.

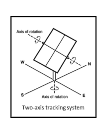

Fig. 9. Example of a figure caption.

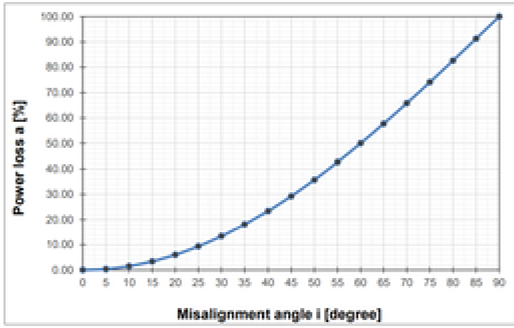

Fig. 10. Example of a figure caption.

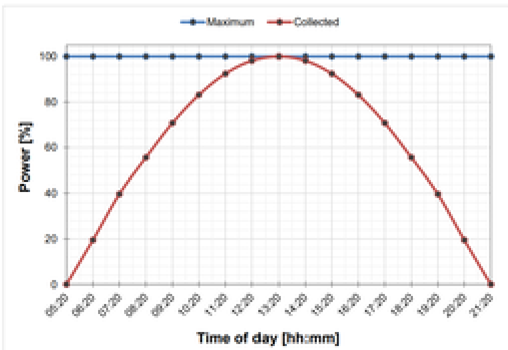

Fig. 11. Example of a figure caption.

XIII. OBSERVATIONS AND RESULT

In this Dual Axis Solar Tracker, when source light falls on the panel, the panel adjusts its position according to maximum intensity of light falling perpendicular to it. The objective of the project is completed. This was achieved through using light sensors that are able to detect the amount of sunlight that reaches the solar panel. The values obtained by the LDRs are compared and if there is any significant difference, there is actuation of the panel using a servo motor to the point where it is almost perpendicular to the rays of the sun.

This was achieved using a system with three stages or subsystems. Each stage has its own role. The stages were;
An input stage that was responsible for converting incident light to a voltage.
A control stage that was responsible for controlling actuation and decision making.
A driver stage with the servo motor. It was responsible for actual movement of the panel.

The input stage is designed with a voltage divider circuit so that it gives desired range of illumination for bright illumination conditions or when there is dim lighting. The potentiometer was adjusted to cater for such changes. The LDRs were found to be most suitable for this project because their resistance varies with light. They are readily available and are cost effective. Temperature sensors for instance would be costly.

The control stage has a microcontroller that receives voltages from the LDRs and determines the action to be performed. The microcontroller is programmed to ensure it sends a signal to the servo motor that moves in accordance with the generated error.

The final stage was the driving circuitry that consisted mainly of the servo motor. The servo motor had enough torque to drive the panel. Servo motors are noise free and are affordable, making them the best choice for the project.

XIV. SPECIFICATIONS OF THE HARDWARE REQUIREMENT

A. *FEATURES OF ARDUINO UNO*

1) Microcontroller: ATMEGA 328P

The Atmel®picoPower®ATmega328/P is a low-power CMOS 8-bitmicrocontroller based on the AVR® enhanced RISC architecture. FEATURES:

High Performance, Low Power Atmel®AVR® 8-Bit Microcontroller Family
1. Advanced RISC Architecture

2. 131 Powerful Instructions

3. Most Single Clock Cycle Execution

4. 32 x 8 General Purpose Working Registers

5. Fully Static Operation

6. Up to 20 MIPS Throughput at 20MHz

7. On-chip 2-cycle Multiplier

8. High Endurance Non-volatile Memory Segments

9. 32KBytes of In-System Self-Programmable Flash program Memory
10. 1KBytes EEPROM

11. 2KBytes Internal SRAM

12. Write/Erase Cycles: 10,000 Flash/100,000 EEPROM
13. Data Retention: 20 years at 85°C/100 years at 25°C(1)

14. Optional Boot Code Section with Independent Lock Bits

15. In-System Programming by On-chip Boot Program

16. True Read-While-Write Operation
17. Programming Lock for Software Security

18. Operating Voltage: 5v
19. Input Voltage (recommended): 7-12V
20. Input Voltage (limits): 6-20V
21. Digital I/O Pins: 14 (of which 6 provide PWM output)
22. Analog Input Pins: 6
23. DC Current per I/O Pin: 40 mA
24. DC Current for 3.3V Pin: 50 mA
25. Flash Memory: 32 KB of which 0.5 KB used by bootloader
26. SRAM: 2 KB (ATmega328)
27. EEPROM: 1 KB (ATmega328)
28. Clock Speed: 16 MHz

B. SOLAR PANEL

1) Maximum Voltage: 4volts (under load)
2) Maximum Voltage: 4.8volts (no load)
3) Rated Current: 100mA
4) Dimension: 6 cm (L) x 6 cm (W) x 0.25 cm (t)
5) Maximum Wattage: 0.5W

C. Servo Motor

Parameter	Modulation Analog
Torque (4.8V)	3.17 kg-cm
Torque (6.0V)	4.10 kg-cm
Speed (4.8V)	0.23 sec/60°
Speed (6.0V)	0.19 sec/60°
Weight	37.2 g
Dimensions	39.9mm x 20.1mm x 36.1mm
Motor Type	3 pole Ferrite
Gear Type	Plastic
Rotation/Support	Bushing
Operating Angle	45 Deg. one side pulse travelling
Pulse Cycle	30 ms
Pulse Width	500-300 µs
Connector Type	J

TABLE I
MODULATION ANALOG SPECIFICATIONS

REFERENCES

[1] Prinsloo, Gerro Dobson, Robert. (2015). Solar Tracking eBook PrinslooDobson2015.
[2] Zhan, Tung-Sheng, et al. "Design and implementation of the dual-axis solar tracking system." 2013 IEEE 37th Annual Computer Software and Applications Conference. IEEE, 2013.
[3] Jamroen, Chaowanan, et al. "A low-cost dual-axis solar tracking system based on digital logic design: Design and implementation." Sustainable Energy Technologies and Assessments 37 (2020): 100618.
[4] Othman, Nashaat, et al. "Performance analysis of dual-axis solar tracking system." 2013 IEEE International Conference on Control System, Computing and Engineering. IEEE, 2013.
[5] Sidek, M. H. M., et al. "Automated positioning dual-axis solar tracking system with precision elevation and azimuth angle control." Energy 124 (2017): 160-170..
[6] Yao, Yingxue, et al. "A multipurpose dual-axis solar tracker with two tracking strategies." Renewable Energy 72 (2014): 88-98.