Department of Electrical Engineering and Computer Science (DEECS)

Solar Panel Tracking Sun for Maximum Efficiency

Present by

Yuexuan Zhang

Mohan Yu

Yubo Cheng

April 2022

Submitted in partial fulfillment of the degree of

Bachelor of Science in

Electrical Engineering

Project Advisor: Massoud Maxwell Rabiee

DEDICATION

We would like to express our gratitude to all those who helped us during the process of finishing this senior design project.

Our sincere and hearty thanks and appreciations go firstly to our advisor, Dr. Massoud Maxwell Rabiee, whose suggestions and encouragement have given us much insight into the studies in this field. It has been a great privilege and joy to study under his guidance and supervision. Furthermore, it is our honor to benefit from his personality and diligence, which we will treasure our whole life. Our gratitude to him knows no bounds.

We are also extremely grateful to all the other faculty members of the Department of Electrical Engineering and Computer Science for their patient instructions in various courses and their precious suggestions for our study here.

Finally, we are grateful to all those who devote much time to reading this thesis and giving us advice, which will benefit us in our later studies.

CONTENTS

LIST OF FIGURES

LIST OF TABLES

ABSTRACT

The system is a solar panel that rotates simultaneously to follow the sunlight. The part of the system that senses changes in light is through four LDRs mounted on the four sides of the solar panel. There are two DC motors to drive the solar panel's movement. The control system is based on Arduino UNO. In future improvements, the platform could be modified, and display modules will be added to the system to show the output power of the solar panel.

INTRODUCTION

The project completed a model of a solar panel that can rotate with light changing. The purpose of this prototype is to maximize solar power efficiency and replace those using non-renewable energy sources to generate electricity methods. This document covers the following: Problem, Solution, and Implementation. The project will encompass the materials and knowledge the team has learned in class and while on Co-op. This project allows the team to master hardware assembly, circuit design, programming, and project management.

Problem

The energy shortages as well as the deterioration of the environment facing the world today are prompting the search for an alternative energy source to supplement traditional fossil fuels[2], such as petroleum. These alternative sources include solar, wind, nuclear, hydro, and geothermal, which have all been utilized with varying levels of success [3].

Solar energy is the energy produced using the energy radiated by the sun. It is the cleanest energy source with the least pollution to the climate. Due to the abundant and clean sources of solar energy, more attention has been paid to it. Theoretically, to ensure maximum efficiency, the solar panel and sunlight should be maintained at 90 degrees, which means the angle of incidence need as close to 0° as possible. This can be achieved by tilting the solar panels to face the sun continuously in real-time during the day.

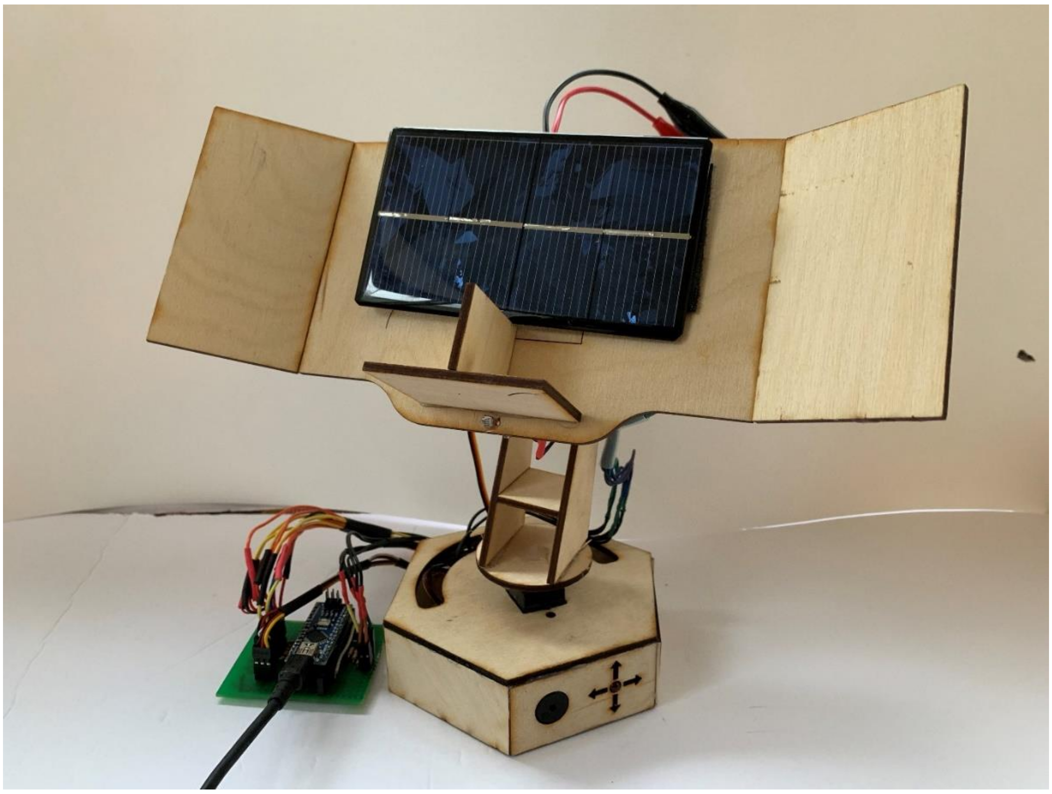


Figure 1: Dual axis solar tracker[1]

Solution

At present, the commonly used solar tracking control methods mainly include the uniform speed control method, light intensity control method, and space-time control method [4]. The solar panel is driven by a dual-axis motor to rotate. “Dual-axis” means that the automated system can track two axes simultaneously, the horizontal (azimuth) and vertical (altitude) directions. Using the dual-axis automatic tracking system, the photovoltaic power generation is expected to be increased by more than 30% [5]. The system can be applied in engineering projects, increasing the efficiency and the revenue of photovoltaic power stations

Uniform speed control

Since the rotation speed of the Earth is fixed, it can be considered that the sun rises from the east in the morning and moves west, and sets in a due south direction. The sun moves at a uniform speed of $15°/h$ in azimuth and moves one week in 24 hours. The altitude angle is equal to the local latitude invariant as a polar axis. The tracking process is that the solar panel fixed on the polar axis rotates at the speed of $15°/h$ of the Earth's rotation, to track the sun and keep the solar panel plane perpendicular to the sunlight.

This method is simple to control, but it is difficult to install and adjust, and the initial Angle is difficult to determine and adjust.

Space-time control

The sun's orbit is dependent on complex factors such as time of day, season, local latitude, and longitude. For tracking the position of the sun rapidly, the supervisory and control program should first calculate the theoretical altitude angle and azimuth angle as coarse adjustments of the automatic tracking mechanism through the following equations [6].

$$\theta_z = cos^{-1}[sin\delta \cdot sin\varphi + cos\delta cos\varphi cos\omega] \qquad (1)$$

$$\alpha = 90° - \theta_z \qquad (2)$$

$$A = cos^{-1}\left[\frac{sin\alpha \cdot sin\delta - sin\delta}{cos\alpha \cdot cos\varphi}\right] \qquad (3)$$

where

θ_z: incidence angle $\quad$ δ: declination

α: altitude angle $\quad$ φ: local latitude

ω: hour angle $\quad$ A: azimuth angle

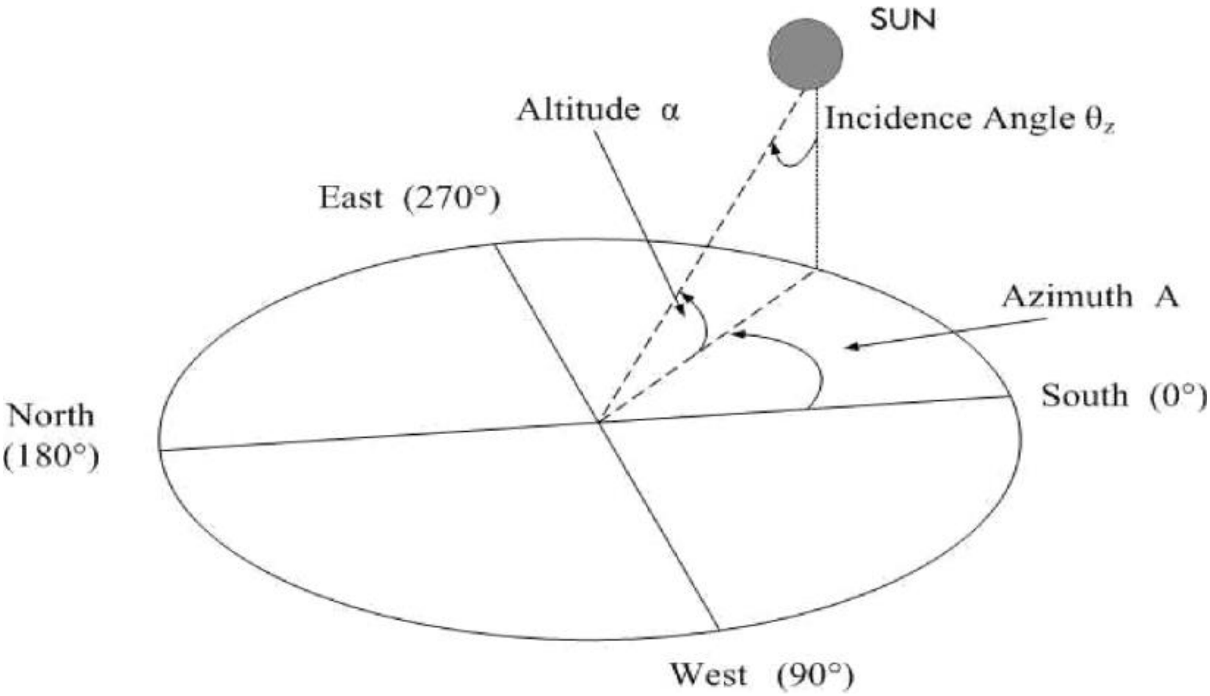


Figure 2: Illustration of Azimuth and Altitude angle of the Sun [6]

Therefore, the above-related data can be pre-input into the microprocessor and calculate the solar azimuth and altitude angle, to achieve synchronization in time and space, and finally get the actual angle to achieve accurate control.

This method has high precision and good adaptability, but the program is complicated and difficult to implement.

Light intensity control

The two LDRs are used as sun position sensing elements for altitude and azimuth tracking, respectively. Four LDRs are fixed in a transparent glass tube. Each pair of photosensitive units is separated by a middle partition and placed symmetrically on both sides of the partition.

When the panel is aligned with the sun, the sunlight is parallel to the partition, the exposure of the two photosensitive units is the same, and the output voltage is the same. When the sunlight is slightly offset, the shadow of the partition will fall on one of the photosensitive units, so that the two photosensitive units have different light amounts and different output voltages. Solar tracking control is performed according to the change of output voltage.

This method has the characteristics of high measurement accuracy, simple circuit, and easy implementation, but it cannot be tracked in a cloudy environment.

Team's choice

After comprehensively considering tracking accuracy, implementation difficulty, student ability, and budget, the team chose the light intensity control method with some to modify as their implementation method.

The team use two groups of LDRs to detect the light change. In each group, there are two LDRs and they are installed on the four edges of the solar panel. The driving part uses two DC motors to control the rotation in the horizontal and vertical directions, respectively. The signal that drives the motor to rotate is generated by the voltage difference of each group of LDRs. The code of the control system is realized by Arduino programming.

Credibility

The three students, Yuexuan Zhang, Mohan Yu, and Yubo Cheng, are all pursuing a Bachelor of Science degree in Electrical Engineering at the University of Cincinnati.

Yuexuan has a solid foundation in embedded system design, electronics, and programming. She gained theoretical and hands-on experiences throughout the related courses, co-op experience in Siemens (Beijing), and competitions. She performed well in her Network Analysis, Electronics, and Embedded Systems courses. During her co-op at Siemens, she was responsible for writing and modifying the program to test the PCB board which improved her programming ability and consolidated her knowledge of electronics. She attended TI Cup National College Student Electronic Design Competition and SCM (single chip machine) Application Design Competition where she gained abundant hands-on experiences. In SCM Application Design Competition, her team designed a vehicle driver health monitoring system based on ECG signal using an AD8232 chip and HT32F52352 single-chip microcontroller. She successfully applied her background knowledge and skills in this project.

Mohan mastered the basics of models, embedded systems, signals, and systems, coding in Arduino through undergraduate study. The co-op experience also gave her a chance to develop hands-on skills for technical work. She joined lab research to design a 38kHz~42kHz equal ripple digital bandpass filter based on MATLAB; during her 16-month internship at the Research Institute of Electronics Technology, she employed MATLAB in mapping and signal transformation and applied Qt to write an upper computer program. These practices let her gain knowledge of communication modules and programming. She also participated in TI Cup National College Student Electronic Design Competition and SCM (single chip machine) Application Design Competition with Yuexuan. Using an AD8232 device and an HT32F52352 single-chip microcontroller, her team created a car driver health monitoring system based on ECG signals in SCM Application Design Competition. These competitions gave her a deeper understanding of microcontrollers. Therefore, all these valuable experiences enable her to complete the task of this project.

Yubo has a solid foundation in embedded system design, electronics, and programming as well. He gained theoretical throughout the related courses at both Chongqing University and the University of Cincinnati. He performed well in his Embedded System, Digital Design, and programming courses. During the summer vacation of 2019, he joined the EECS Robotics Research Program at MIT, where he had a chance to put what he has learned into practice and gained hands-on experiences. During this program, he acquired basic knowledge of mechanical design, electrical components, and software design, but also, he was able to put what he had learned into hands-on practice by assembling an autonomous vehicle using MIT's Mobile Robot Platform.

Project Goals

There are several goals have been identified. The main goal is to make a prototype that the solar panel can rotate in tandem with the sun's movements. With environmental problems and resource shortages, countries are increasing the proportion of new energy generation. And there are many kinds of solar power solutions out there. Part of the research in this project is how to control the solar panels to rotate simultaneously with the sun. This prototype implements this goal through LDRs to detect the light intensity difference with each side to drive the solar panel. This prototype can meet the expectation, but some parts can be improved.

Future improvements aim to replace motors and detection systems, as well as a larger solar panel so that the entire prototype can be used for individuals, or even for industrial power generation.

DISCUSSION

Project Concept

With the increasing depletion of fossil energy and its increasingly severe impact on the environment, people are paying more and more attention to the application of renewable energy to relieve the dependence on fossil energy such as coal. Among all categories of renewable energy, solar energy is clean and abundant. Solar energy has developed rapidly in recent years. However, statistics show that the highest conversion efficiency of solar photovoltaic panels is only 25% [7]. In practical applications, solar photovoltaic panels are fixed. The angle of the solar photovoltaic panels cannot be adjusted, which leads to a further decrease in the utilization rate of solar energy. Therefore, how greatly improving the utilization rate of solar energy is an urgent problem to be solved.

This project proposed a solar tracking system that can keep the plane of the solar panel perpendicular to the incident angle of the sunlight. Therefore, the photoelectric conversion efficiency of the solar panel can be improved. The concept map on the next page shows the background, objective, and implementation of this project and also team members' knowledge and skills associated with this project.

The hardware includes an Arduino UNO board, a solar panel, 4 light-dependent resistors, two servo motors controlling horizontal and vertical movement, and a tilt to hold the solar panel and motors. The light-dependent resistors placed on four edges of the solar panel are in four voltage division circuits. When the light intensity changes, the resistance of the light-dependent resisters will change. The voltages divided on each light-dependent resistor will also change. Therefore, the light intensity is converted to electrical signals by the light detection circuit. The microcontroller calculates the voltage difference of the light-dependent resistors on the opposite edges. When the sunlight is perpendicular to the solar panel, there should be no difference between the voltages of the light-dependent resistors on the opposite edge. If there is a difference, the microcontroller will send instructions to the horizontal servo and vertical servo to adjust the orientation and angle of the solar panel.

For the software, we used Arduino IDE to write the code. The algorithm and code will be discussed in detail in the Methodology part.

We have applied our math and physics knowledge, co-op experience, public speaking, and technical writing skills to this project. This project also requires the background knowledge from the following courses: Fundamentals of Engineering, Network Analysis, Electronics, and Embedded System.

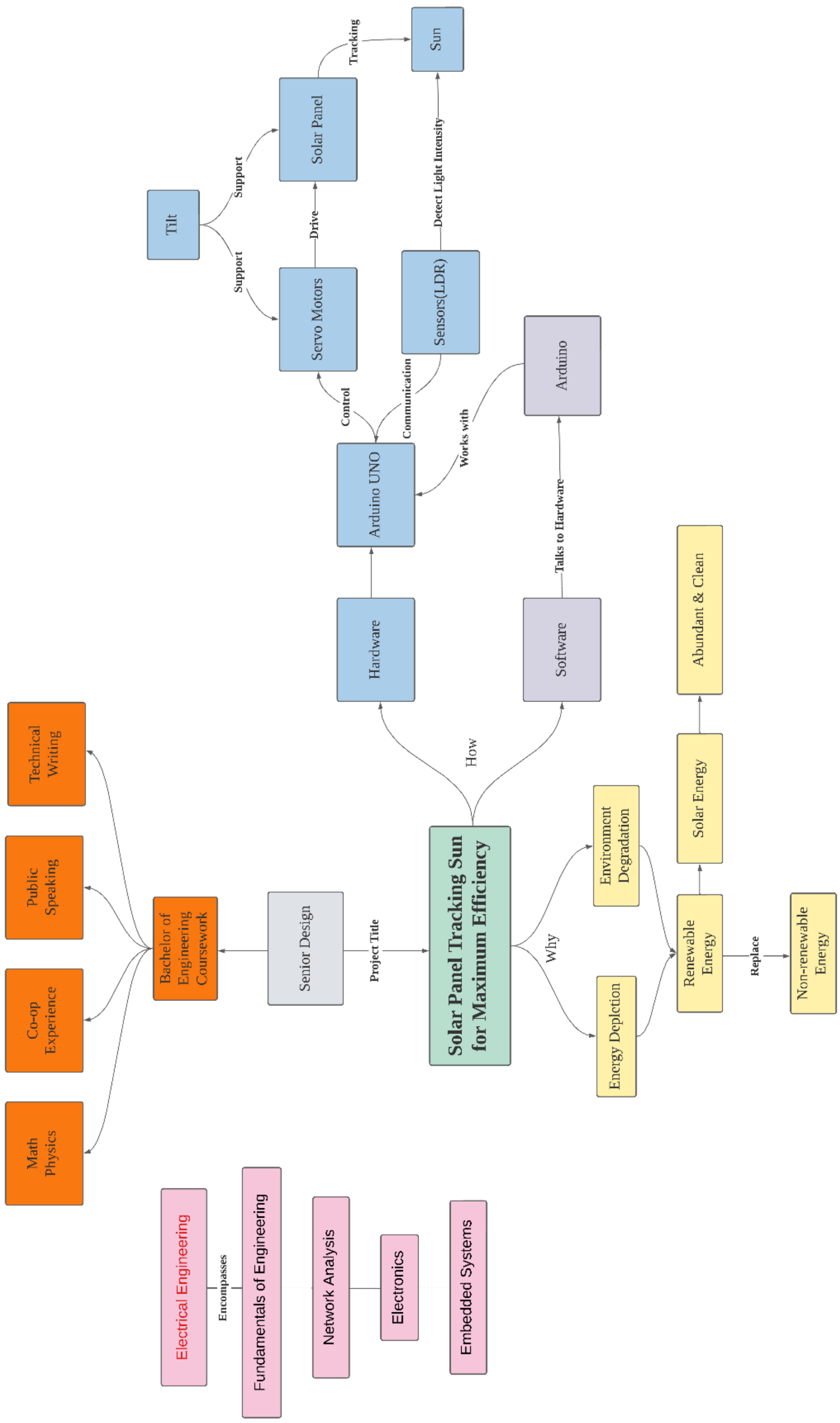


Figure 3: Concept Map

Design Objectives

To achieve automatic tracking of sunlight, our step-by-step objectives are:

- ✓ Two servos control the horizontal and vertical movement of the solar panel.
- ✓ The reaction time of servos is less than 1 second.
- ✓ Solar panel with the vertical incidence of sunlight in real-time.
- ✓ Position information of the solar panel is sent to Arduino as feedback.
- ✓ Arduino microcontroller for processing signals and sending instructions to the servos.

Methodology

Control method

Based on the test set, sensitivity, and cost, we choose to adopt the light intensity control approach. This method is easier to test once the sample has been assembled, and has more reference materials. To implement automatic tracking control, the connection between each part of the prototype is shown in the figure.

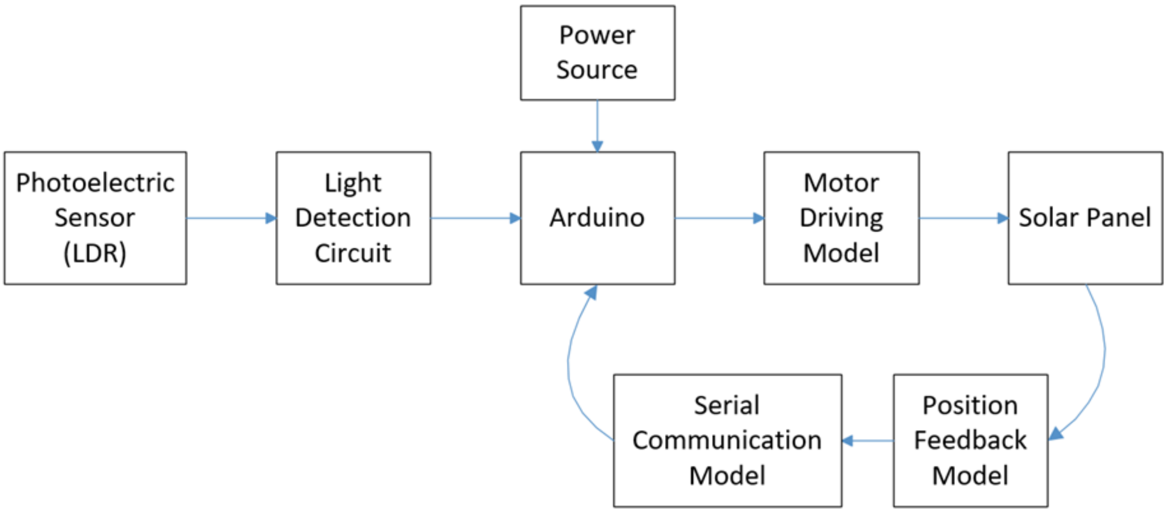


Figure 4: Structure of Solar Panel Tracking Sun System

The system mainly consists of Arduino UNO, the photoelectric sensor is LDR (Light Dependent Resistor), Servos (SG90), and three extra power sources. Sunlight is converted into electrical signals by LDR placed on the horizontal and vertical axis of the solar panel based on Cartesian coordinates and then is input into the microcontroller. After processing and calculation, the microcontroller sends instructions to the horizontal servo and vertical servo on the mini platform to control the orientation and angle of the solar panel in real-time and achieve the purpose of the automatic tracking of sunlight.

Hardware

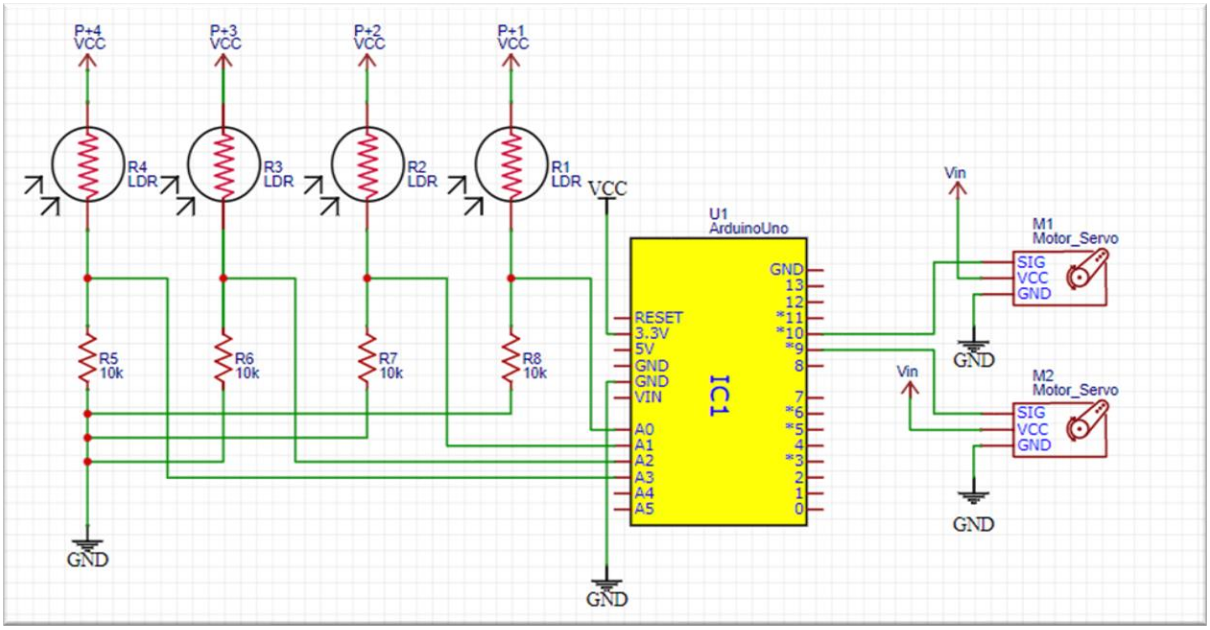


Figure 5: Schematic Diagram of hardware

LDR converts sunlight information into an electrical signal. Light Dependent Resistor is a special resistor made of semiconductor photoconductive effect, which is very sensitive to light. With the increase of light intensity, its resistance decreases rapidly. Because different light intensities lead to different electric potentials of LDR, the server will attempt to move the solar panel to a position where both symmetrical LDRs have the same resistance, which means that the amount of light on both resistors will be the same.

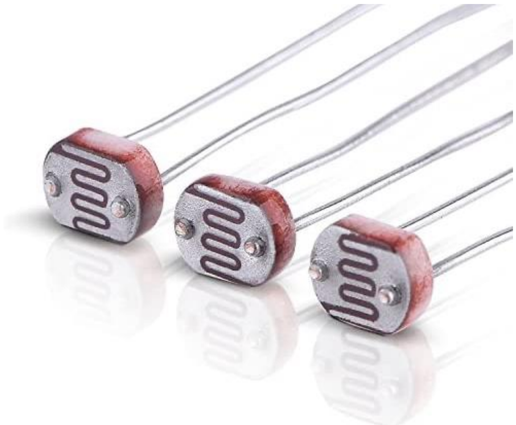


Figure 6: Light-dependent resistor used in this project

Microcontroller - Arduino UNO.

We use it because UNO is cheap and has all the modules we need like ADC. The Arduino UNO is a microcontroller board based on the ATmega328P. There are 14 digital input/output pins, six analog inputs, a 16 MHz ceramic resonator, a USB connection, a power jack, an ICSP header, and a reset button on the board. It can be started by simply plugging it into a computer with a USB cable or powering it with an AC-to-DC adapter or battery (source: https://www.arduino.cc/).

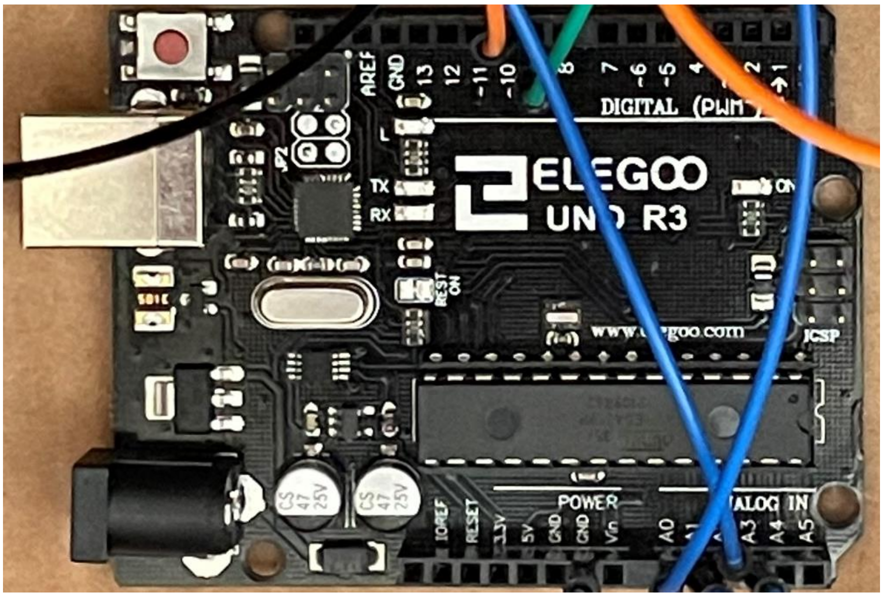


Figure 7: Arduino UNO used in this project

The pins framed in blue are all the pins we used in this project.

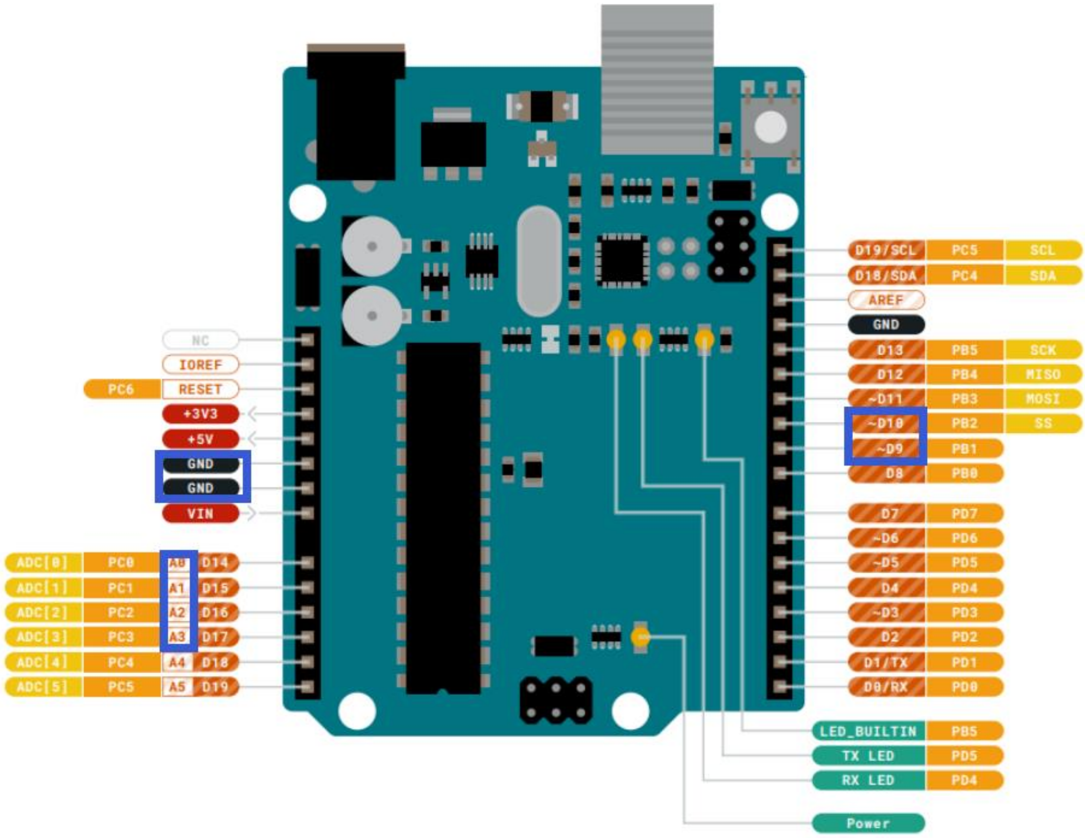


Figure 8: Pinout of Arduino UNO

SG90 to achieve mechanical tracking.

In SG90, the red line is connected to VCC, the brown line is connected to GND, and the orange line is connected to the signal. Its working voltage is 4.2V ~ 6V. The control signal of the steering gear is a pulse width modulation (PWM) signal with a period of 20ms, in which the pulse width ranges from 0.5ms-2.5ms and the position of the corresponding rudder plate is 0-180 degrees which changes linearly. There is a reference circuit inside the steering gear, which generates a reference signal with a period of 20ms and width of 1.5ms. To generate the motor's rotation signal, a comparator compares the external signal with the reference signal to assess its direction and size. The control circuit board receives the corresponding PWM control signal from the signal line and then controls the rotation of the motor. The motor drives a series of gear sets and transmits them to the output rudder plate after deceleration. The output shaft of the steering gear relates to the position feedback potentiometer. While the steering wheel rotates, it drives the position feedback potentiometer. The potentiometer will provide feedback to the control circuit board in the form of a voltage signal, and the control circuit board will decide the rotation direction and speed of the motor based on the location, allowing the motor to stop when it reaches the target.

Figure 9: SG90 used in this project

Protective resistors.

We use 10 KΩ resistors as protective resistors to prevent excessive current in the circuit, thus burning out components.

After fully understanding the working principle and characteristics of all components, we assemble them.

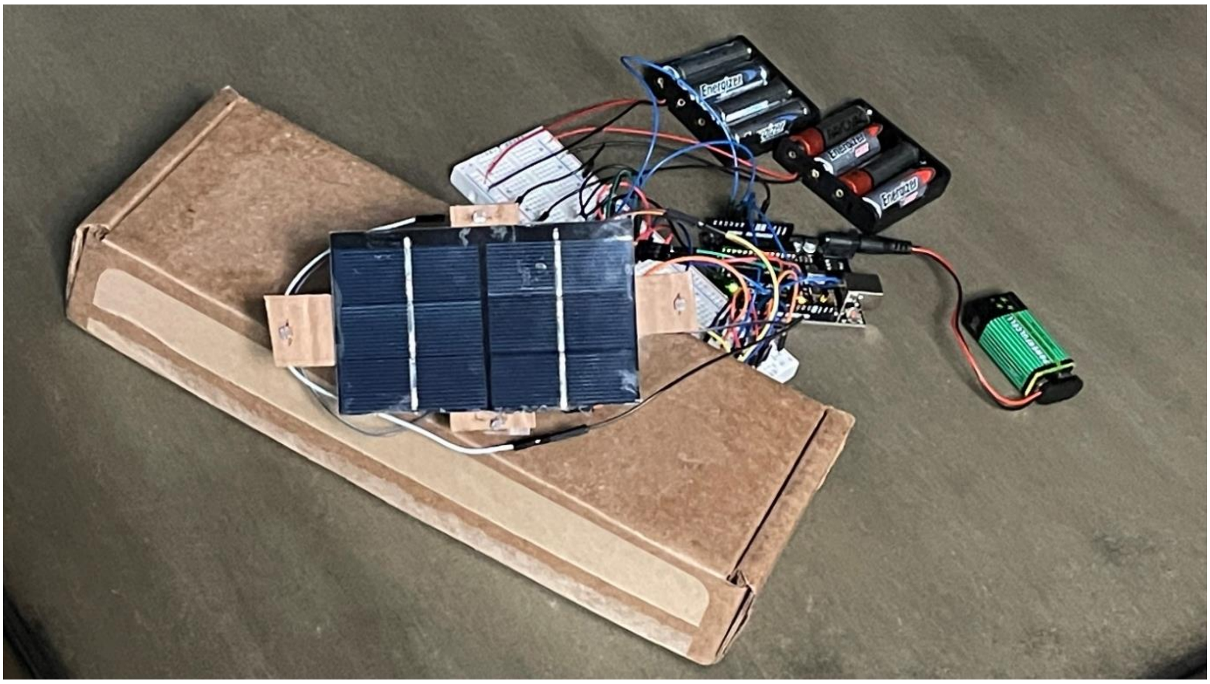


Figure 10: Prototype of Solar Panel Tracking Sun System

Software

Dual-axis automatic tracking system controlled by Arduino code

The main part of the code is divided into the following four steps:

1. Include servo Library.

```
#include <Servo.h>
```

2. Assign LDR Pins so that we can read the value from the sensor and move the solar panel accordingly.

```
int topLDRpin = A0;   //A0 for the LDR on top position
int leftLDRpin = A1;   //A1 for the LDR on top position
int bottomLDRpin = A2;   //A2 for the LDR on top position
int rightLDRpin = A3;   //A3 for the LDR on top position
```

3. Initialize servo position.

```
int topLDR = 0;
int rightLDR = 0;
int leftLDR = 0;
int bottomLDR = 0;
```

4. Use the loop to adjust the servo motion through the difference in LDR

```
void loop()
{
  //Read and print the values of the LDR sensors
  rightLDR = analogRead(rightLDRpin);
  leftLDR = analogRead(leftLDRpin);
  topLDR = analogRead(topLDRpin);
  bottomLDR = analogRead(bottomLDRpin);
  Serial.print("rightLDR"); Serial.print(rightLDR); Serial.print(" ");
  Serial.print("leftLDR");Serial.print(leftLDR);Serial.print(" ");
  Serial.print("topLDR");Serial.print(topLDR);Serial.print(" ");
  Serial.print("bottomLDR");Serial.print(bottomLDR);Serial.print(" ");

  horizontalError = rightLDR - leftLDR;   //Difference between the left and sensors

  if(horizontalError>20)   //If the error is greater than 20 then move the solar panel to the
right
  {
      servoHPos--;
      servoHPos = constrain (servoHPos, 0,179);
      servoH.write(servoHPos);
```

```
  }
  else if(horizontalError<-20)   //If the error is less than -20 then move the solar panel to the left
  {
       servoHPos++;
       servoHPos = constrain (servoHPos, 0,179 );
       servoH.write(servoHPos);
  }
  verticalError = topLDR - bottomLDR; //Difference between top and bottom sensors
  if(verticalError>20)   //If the error is greater than 20 then move the solar panel up
  {
       servoVPos++;
       servoVPos = constrain (servoVPos, 0,179);
       servoV.write(servoVPos);
  }
  else if(verticalError<-20)   //If the error is less than -20 then move the solar panel down
  {
       servoVPos--;
       servoVPos = constrain (servoVPos, 0,179);
       servoV.write(servoVPos);
  }
 delay(25);
}
```

The complete code is in the appendix.

Budget

The table below shows the budget of this project. All the parts and supplies are purchased online. The overall cost of this design is low, which is one of its advantages of this design.

Parts	Price	Quantity	Total cost
Arduino Uno Project Kit	$38.99	1	$38.99
2W 6V Solar Panel	$7.99	1	$7.99
Mini Pan/Tilt Camera Platform with two SG 90 Servos	$11.99	1	$11.99
Light-dependent resistor	$0.17	30	$5.15
4 x AA Thicken Battery Holder	$1.50	4	$5.99
AA Batteries	$1.63	4	$6.49
		Project Total:	**$76.60**

Table 1: Project Budget

Timeline

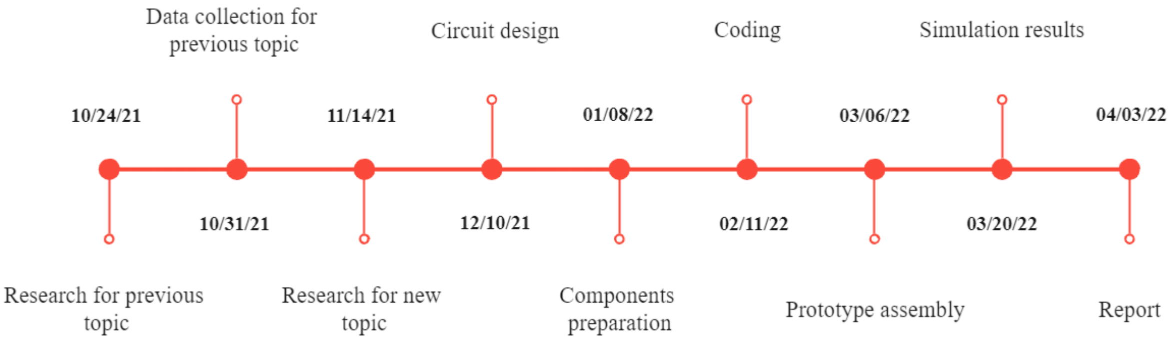
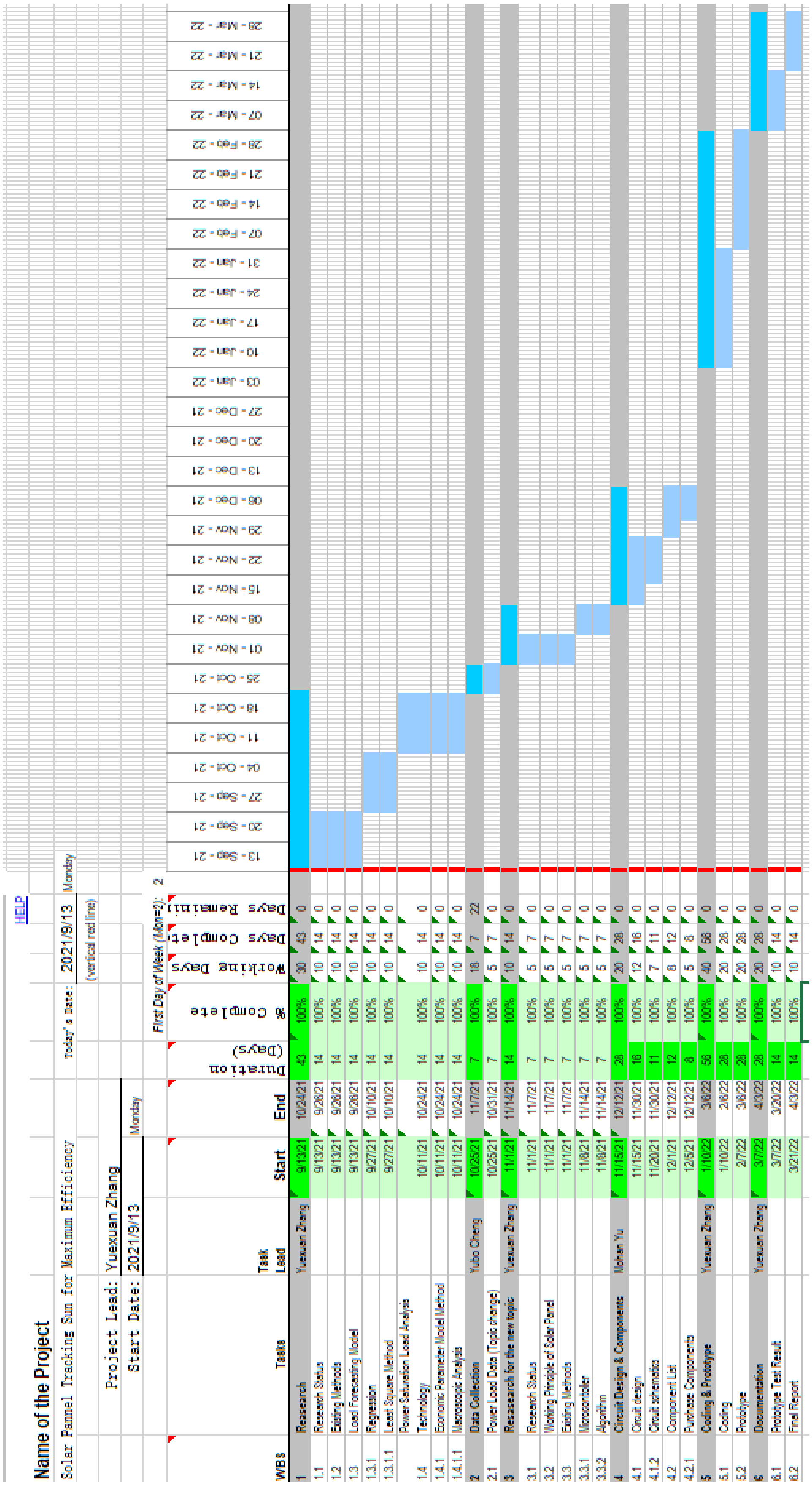


Figure 11: Timeline of Senior Design Project

In the fall semester, we finished topic research and circuit design. We studied the research status of solar panel control, working principle of solar panel, the existing solar panel control methods, different kinds of microcontrollers and corresponding algorithms. We finally decided to use Arduino as the microcontroller for its flexibility and useful functions. We finished components preparation before the beginning of the spring semester. Then we spent one month on programming and one and a half month on prototype assembly and testing. Finally we finished the whole project according to our plan.

The more specific time allocation of each part of the project is shown in the Gantt Chart on the next page.



HELP

Name of the Project

Solar Pannel Tracking Sun for Maximum Efficiency

Today's Date: 2021/9/13 Monday

(vertical red line)

Project Lead: Yuexuan Zhang

Start Date: 2021/9/13 Monday

First Day of Week (Mon=2): 2

WBS	Tasks	Task Lead	Start	End	Duration (Days)	% Complete	Working Days	Days Complete	Days Remaining
1	Research	Yuexuan Zhang	9/13/21	10/24/21	43	100%	30	43	0
1.1	Research Status		9/13/21	9/26/21	14	100%	10	14	0
1.2	Existing Methods		9/13/21	9/26/21	14	100%	10	14	0
1.3	Load Forecasting Model		9/13/21	9/26/21	14	100%	10	14	0
1.3.1	Regression		9/27/21	10/10/21	14	100%	10	14	0
1.3.1.1	Least Square Method		9/27/21	10/10/21	14	100%	10	14	0
1.4	Power Saturation Load Analysis Technology		10/11/21	10/24/21	14	100%	10	14	0
1.4.1	Economic Parameter Model Method		10/11/21	10/24/21	14	100%	10	14	0
1.4.1.1	Macroscopic Analysis		10/11/21	10/24/21	14	100%	10	14	0
2	Data Collection	Yubo Cheng	10/25/21	11/7/21	7	100%	18	7	22
2.1	Power Load Data (Topic change)		10/25/21	10/31/21	7	100%	5	7	0
3	Resasearch for the new topic	Yuexuan Zhang	11/1/21	11/14/21	14	100%	10	14	0
3.1	Research Status		11/1/21	11/7/21	7	100%	5	7	0
3.2	Working Principle of Solar Panel		11/1/21	11/7/21	7	100%	5	7	0
3.3	Existing Methods		11/1/21	11/7/21	7	100%	5	7	0
3.3.1	Microcontoller		11/8/21	11/14/21	7	100%	5	7	0
3.3.2	Algorithm		11/8/21	11/14/21	7	100%	5	7	0
4	Circuit Design & Components	Mohan Yu	11/15/21	12/12/21	28	100%	20	28	0
4.1	Circuit design		11/15/21	11/30/21	16	100%	12	16	0
4.1.2	Circuit schematics		11/20/21	11/30/21	11	100%	7	11	0
4.2	Component List		12/1/21	12/12/21	12	100%	8	12	0
4.2.1	Purchase Components		12/5/21	12/12/21	8	100%	5	8	0
5	Coding & Prototype	Yuexuan Zhang	1/10/22	3/6/22	56	100%	40	56	0
5.1	Coding		1/10/22	2/6/22	28	100%	20	28	0
5.2	Prototype		2/7/22	3/6/22	28	100%	20	28	0
6	Documentation	Yuexuan Zhang	3/7/22	4/3/22	28	100%	20	28	0
6.1	Prototype Test Result		3/7/22	3/20/22	14	100%	10	14	0
6.2	Final Report		3/21/22	4/3/22	14	100%	10	14	0

Figure 12: Gantt Chart of Senior Design Project

Problem Encountered/Analysis of Problem Solved

The power supply of the servo motors

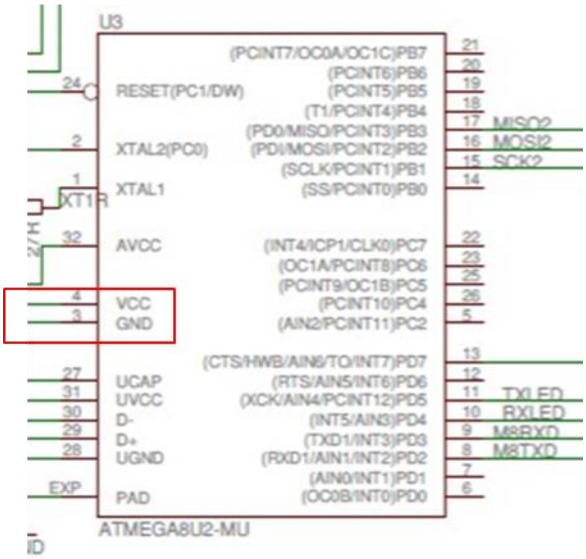


Figure 13: Part of Arduino UNO Schematic

As shown in the above figure, Arduino UNO only has one 5V output voltage port. The operating voltage range of SG90 servo motors is 4.8~6V. The Arduino board can’t power two servo motors at the same time. At the beginning of the prototype test, we tried to use Arduino UNO to power one servo motor and use the battery to power the other one. Then we found it was not ideal to use Arduino UNO to power even one servo motor. The motor often shakes. Finally, we decided to use batteries (4 AA batteries in a battery slot for each servo motor) to power both servo motors. The following figure shows the power supply of the servo motors.

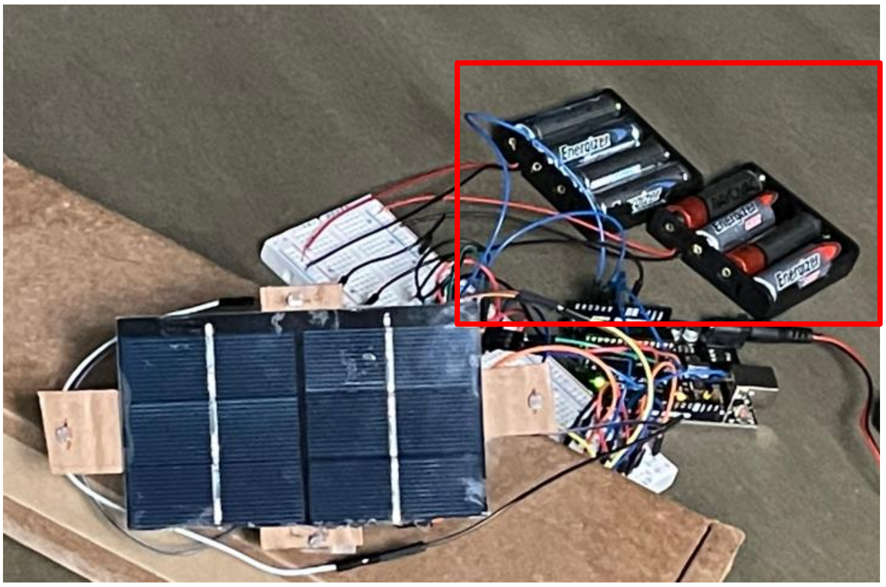


Figure 14: Power Supply of Servo Motors

Installation of the solar panel

The tilt we bought was designed for holing cameras and there were protruding parts that prevented us from installing the solar panel on the tilt. Finally, we cut the parts in the red circle off.

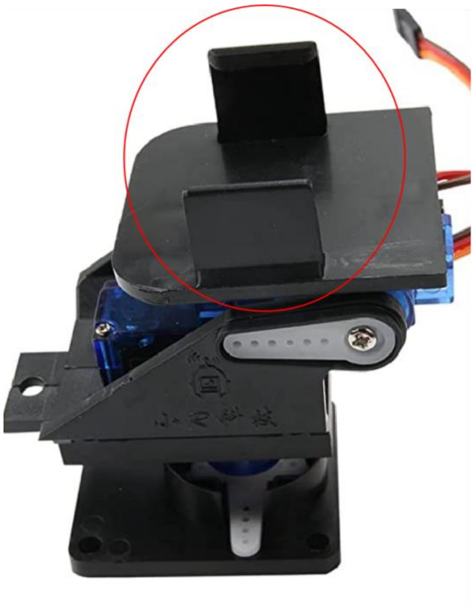


Figure 15: Tilt to Hold the Solar Panel and Servo Motors

Increase the stability of the entire tracking system

The weight of the tilt is relatively light compared to the solar panel, which causes the instability of the whole system. To solve this problem, we added a base and fixed the til on it, and cut off half of the solar panel to lighten its weight. After those modifications, the tracking system became more stable and accurate.

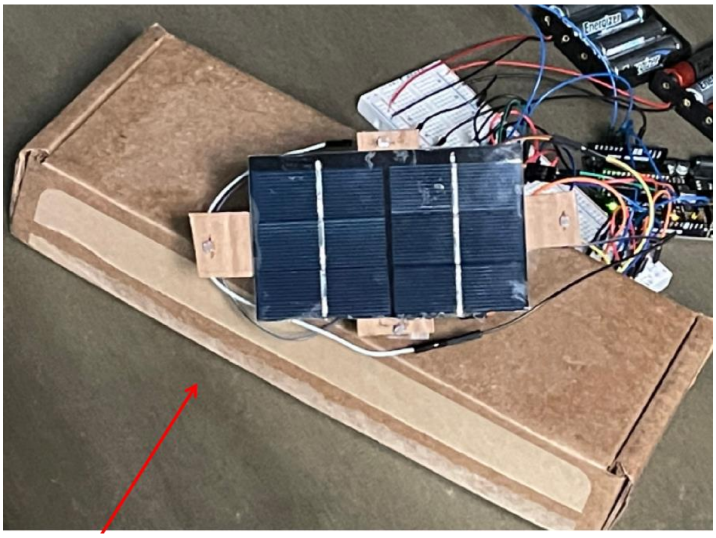


Figure 16: The Base of the Tilt

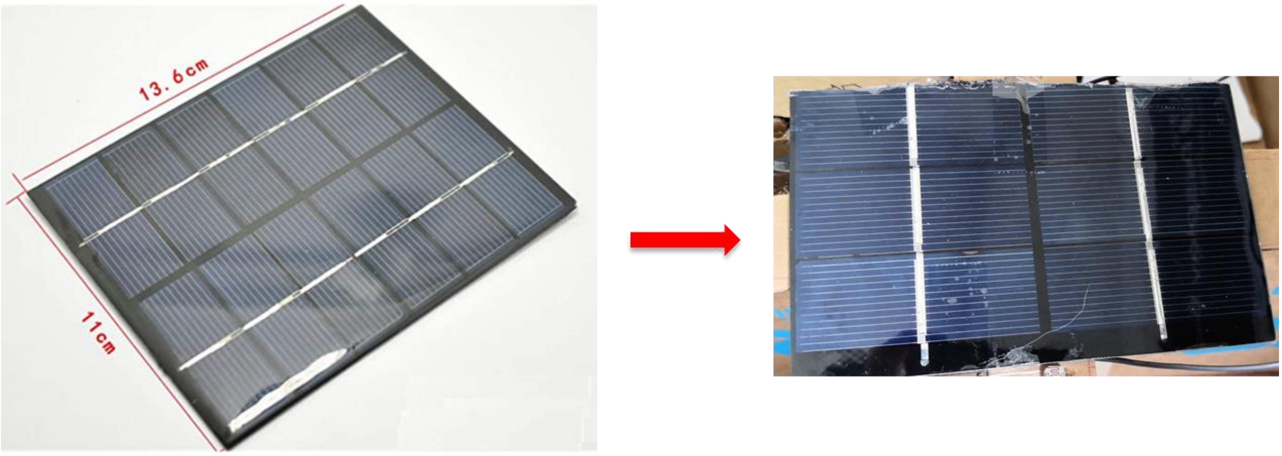


Figure 17: Solar Panel

Future Recommendations

The algorithm of this design is concise, and the mechanical structure is simple. The accuracy and sensitivity are satisfying. This design has the potential to be scaled up and applied to solar power plants or home solar generators.

The solar tracking system can greatly improve the utilization rate of solar energy. The dual-axis solar tracking can track the changes in the azimuth and altitude of the sun at the same time. Theoretically, dual-axis tracking can track the sun's trajectory, always keeping sunlight perpendicular to the solar panel. Compared to the single-axis solar tracking system, dual-axis tracking has higher accuracy. In addition to accuracy, designing an efficient solar tracking system should also consider the impact of the external environment, such as climate and terrain. This design has wider application if the sensor can be optimized for different weather (different light intensity).

The platform that holds the solar panel and servos can be modified to be neater and more stable.

A display module could be added to display the real-time output power of the solar panel. Another module that monitors the power consumed by the tracking system can also be added.

CONCLUSION

This project is aimed at improving the efficiency of solar panels. A sun tracking system is designed based on the light-dependent resistor feedback method to drive the solar panel to track the sun in real-time through SG90 servo motors and controlled by Arduino UNO.

The solar panel tracking system in this design mainly consists of an Arduino UNO board, four light-dependent resistors (LDR) as sensors placed on the four edges of the solar panel, and two SG90 servo motors controlling the horizontal and vertical movement of the solar panel. The light-dependent resistors are in four voltage division circuits. When the light intensity changes, the resistance of the light-dependent resisters will change. The voltages divided on each light-dependent resistor will also change. Therefore, the light intensity is converted to electrical signals by the light detection circuit. After processing and calculating the voltage difference of the light-dependent resistors on the opposite edges, the microcontroller sends instructions to the horizontal servo and vertical servo to control the orientation and angle of the solar panel.

The prototype test result shows that the system has high accuracy, and the system response time is less than 1 second.

REFERENCE

[1] OmA17. "Building an Automatic Solar Tracker With Arduino Nano V2." (accessed.

[2] S. Racharla and K. Rajan, "Solar tracking system – a review," *International Journal of Sustainable Engineering,* vol. 10, no. 2, pp. 72-81, 2017/03/04 2017, DOI: 10.1080/19397038.2016.1267816.

[3] J. Rizk and Y. Chaiko, "Solar tracking system: more efficient use of solar panels," *World Academy of Science, Engineering and Technology,* vol. 41, no. 2008, pp. 313-315, 2008.

[4] X. Wang, Y. Wang, J. Yan, W. Zhao, and Z. Zhang, "Design of Automatic Tracking Control System for Solar Panel," (in Chinese), *Journal of Northwest University (Natural Science Edition),* vol. 34, no. 2, p. 2, 2004.

[5] D. Ling and W. Li, "Study on automatic tracking equipment of PV power supply with double photoelectricity sensor," (in Chinese), *Journal of Qinghai University (Nature Science),* vol. 01, pp. 10-13, 2007.

[6] T. Zhan, W. Lin, M. Tsai, and G. Wang, "Design and Implementation of the Dual-Axis Solar Tracking System," *2013 IEEE 37th Annual Computer Software and Applications Conference,* pp. 276-277, 2013, DOI: 10.1109/COMPSAC.2013.46.

[7] X. Zhang, Q. Liu, and H. Ma, "Analysis of Influencing Factors of Output Power of Photovoltaic Power Station," (in Chinese), *Grid and Clean Energy,* vol. 5, pp. 75-81, 2012.

APPENDIX

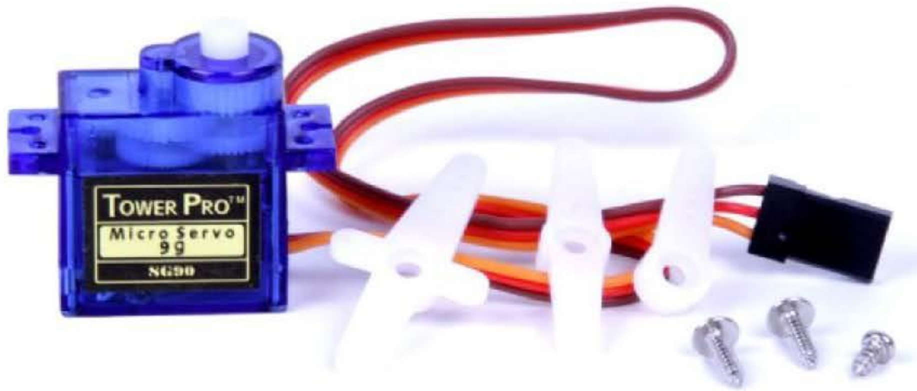
Appendix A: Arduino code

```
/*
*EE Senior Design Project: Solar Panel Tracking Sun for Maximum Efficiency
*Presented by:
*Yubo Cheng
*Mohan Yu
*Yuexuan Zhang
*Date: 10/2/22
*/

  // Servo library

//Servos
Servo servoH;  // Create servo object for the servo that controls horizontal movements
Servo servoV;  // Create servo object for the servo that controls vertical movements

//LDR Pins Assignment:
int topLDRpin = A0;  //A0 for the LDR on top position
int leftLDRpin = A1;  //A1 for the LDR on top position
int bottomLDRpin = A2;  //A2 for the LDR on top position
int rightLDRpin = A3;  //A3 for the LDR on top position


//LDRs initialization
int topLDR = 0;
int rightLDR = 0;
int leftLDR = 0;
int bottomLDR = 0;


//Differences between right/left and top/bottom LDRs
int horizontalError = 0;
int verticalError = 0;

//Create variables to store the servo position
int servoHPos = 0;
int servoVPos = 0;
```

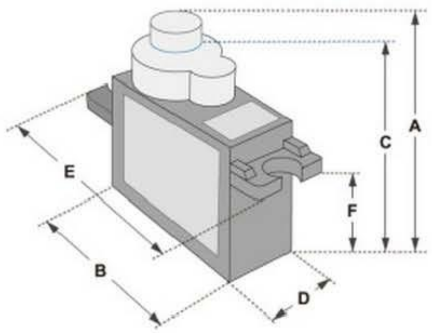
```
void setup()
{
 //Servo pins
 //Pin 9 and 10 are PWM pins
 servoH.attach(9);
 servoV.attach(10);
 Serial.begin(9600);   //Start serial communication, baud rate 9600
}

void loop()
{
  //Read and print the values of the LDR sensors
  rightLDR = analogRead(rightLDRpin);
  leftLDR = analogRead(leftLDRpin);
  topLDR = analogRead(topLDRpin);
  bottomLDR = analogRead(bottomLDRpin);
  Serial.print("rightLDR"); Serial.print(rightLDR); Serial.print(" ");
  Serial.print("leftLDR");Serial.print(leftLDR);Serial.print(" ");
  Serial.print("topLDR");Serial.print(topLDR);Serial.print(" ");
  Serial.print("bottomLDR");Serial.print(bottomLDR);Serial.print(" ");

  horizontalError = rightLDR - leftLDR;  //Difference between the left and sensors

  if(horizontalError>20)  //If the error is greater than 20 then move the solar panel to the right
  {
      servoHPos--;
      servoHPos = constrain (servoHPos, 0,179);
      servoH.write(servoHPos);
  }
  else if(horizontalError<-20)  //If the error is less than -20 then move the solar panel to the left
  {
      servoHPos++;
      servoHPos = constrain (servoHPos, 0,179 );
      servoH.write(servoHPos);
  }


  verticalError = topLDR - bottomLDR; //Difference between top and bottom sensors

  if(verticalError>20)  //If the error is greater than 20 then move the solar panel up
  {
      servoVPos++;
      servoVPos = constrain (servoVPos, 0,179);
      servoV.write(servoVPos);
  }

  else if(verticalError<-20)  //If the error is less than -20 then move the solar panel down
  {
      servoVPos--;
      servoVPos = constrain (servoVPos, 0,179);
      servoV.write(servoVPos);
  }
 delay(25);
}
```

Appendix B: Datasheet of SG 90 Servo

SERVO MOTOR SG90 DATA SHEET



Tiny and lightweight with high output power. Servo can rotate approximately 180 degrees (90 in each direction), and works just like the standard kinds but smaller. You can use any servo code, hardware or library to control these servos. Good for beginners who want to make stuff move without building a motor controller with feedback & gear box, especially since it will fit in small places. It comes with a 3 horns (arms) and hardware.



Dimensions & Specifications
A (mm) : 32
B (mm) : 23
C (mm) : 28.5
D (mm) : 12
E (mm) : 32
F (mm) : 19.5
Speed (sec) : 0.1
Torque (kg-cm) : 2.5
Weight (g) : 14.7
Voltage : 4.8 - 6

Position "0" (1.5 ms pulse) is middle, "90" (~2ms pulse) is middle, is all the way to the right, "-90" (~1ms pulse) is all the way to the left.