NORTH SOUTH UNIVERSITY

Department of Electrical & Computer Engineering

EEE 299: Junior Design Project

Course Instructor: Dr. Shahnewaz Siddique
Section: 02

Name of The Project: Sun Light Tracking Solar Panel with Real Time Power Logger.

Group No: 01

NAME	ID
Md.Rofiqul Islam	1521070643
Nuren Shams	1530784043
Abrar Fahim	1610522043

Abstract

The life without electricity is unimaginable for almost all of us in this world . According to World Bank electric power consumption (KWh per capita) in 2014 was 3127.361. Which gives us an illustration that how much electric power dependent we really are and how vulnerable we are without it. In terms electricity generation, it is generated in many ways. For example by using oil, gas, hydro power, solar power, nuclear power etc . From those sources the solar energy is one of the most abundant sources that human can find. Although there is abundant Sun light and easy to get solar panels but the efficiency of the solar panels are quiet poor. Thus we can harness very little solar energy through regular solar panels. Another thing is that, most of this solar panels are mounted at a fixed angle, thus harnessing solar energy for only couple of hours in the day time. So, one can easily notice the wastage of energy over more wastage. Keeping this problem in our mind, we have designed and build our project in such a way that it can reduce the solar energy wastage as much as possible by us.

Table of Contents

INTRODUCTON:

In the modern world of science and technology renewable and green energy is a buzz word that seems to be uttered constantly. One of the key source of renewable energy comes from harnessing the solar energy from the Sun light. For harnessing solar energy we use solar panels. Most of these solar panels that we used are mounted at a fixed angle. For this type of non-moveable solar panels the user can harness solar energy form the sun at a fixed period of day time . So, rest of the solar energy gets wasted. Thus, keeping this problem in mind we have designed a solution to this energy wastage problem. The solution of this problem is a rotating solar panel module that follows the sun light through the day using LDRs (Light Dependent Resistor) and servo motor as the feedback system, which rotates the module in a vertical and a horizontal direction. In this project we have impelled that solution and also added a measurement device with it which measures the voltage current and also the power delivered by the solar panel to a particular object.

PROBLEM STATEMENT:

The problem that we want to deal is energy wastage. Currently the solar panels that we are using is of very low efficiency which is about 15% to 17%. The max efficiency of solar panels that are being used is about 22.8% (which is SunPowers SPR X21-345 according to solarreviews.com).Thus thinking about this using a fixed angle non-moveable solar panel which waste a lot of solar energy can never be an option.

Now for the challenges that we have faced solving this problem are as follows:

Rotation of the solar panel: Which was done by using the LDRs and servos with the help of an Arduino Uno development board.

The second problem that we have faced building the project is measuring the current that is being supplied to the load by the solar panel. This problem was solved using an INA219 Bi-directional current sensing IC.

LITERATURE REWIEW / TECHNOLOGY REVIEW:

For the technology and literature of this project we consulted multiple websites and youtube tutorials. From those a few of the notable ones are :

- **http://nevonprojects.com/rotating-solar-panel-using-arduino/**
- **https://circuitdigest.com/microcontroller-projects/arduino-solar-panel-tracker**
- **https://www.instructables.com/id/Simple-Dual-Axis-Solar-Tracker/**

The above two projects of rotating solar panel project talks about a single axis rotation of the solar panel but from another literature review given in **https://www.eia.gov/todayinenergy/detail.php?id=18871** we have found that the efficiency of dual axis rotating solar panel is far more than that of a single axis (& single direction facing) rotating solar panel. So, for this project we have chosen the dual axis rotating solar panel although the third project describes the a dual axis rotating solar panel system we have build our own version which uses the Atmega 328p (DataSheet:**https://www.sparkfun.com/datasheets/Components/SMD/ATMega328.pdf**) microcontroller's Timers, Registers and controllers, which also provides a better execution time than the code provided in the last project.

For the power logger part of this project we simply depended on the INA219 Bi-directional current sensing module. For this we needed to review the functionalities of INA219 in the datasheet of the IC(**https://cdn-learn.adafruit.com/downloads/pdf/adafruit-ina219-current-sensor-breakout.pdf**). From there we came to know about the functions: **ina219.getShuntVoltage_mV()** , **ina219.getBusVoltage_V(), ina219.getCurrent_mA()** which were the keys to get the shunt & bus voltage and supplied to the load. And from the shunt and bus voltage we then calculated the load voltage using the formula busvoltage + (shuntvoltage / 1000).

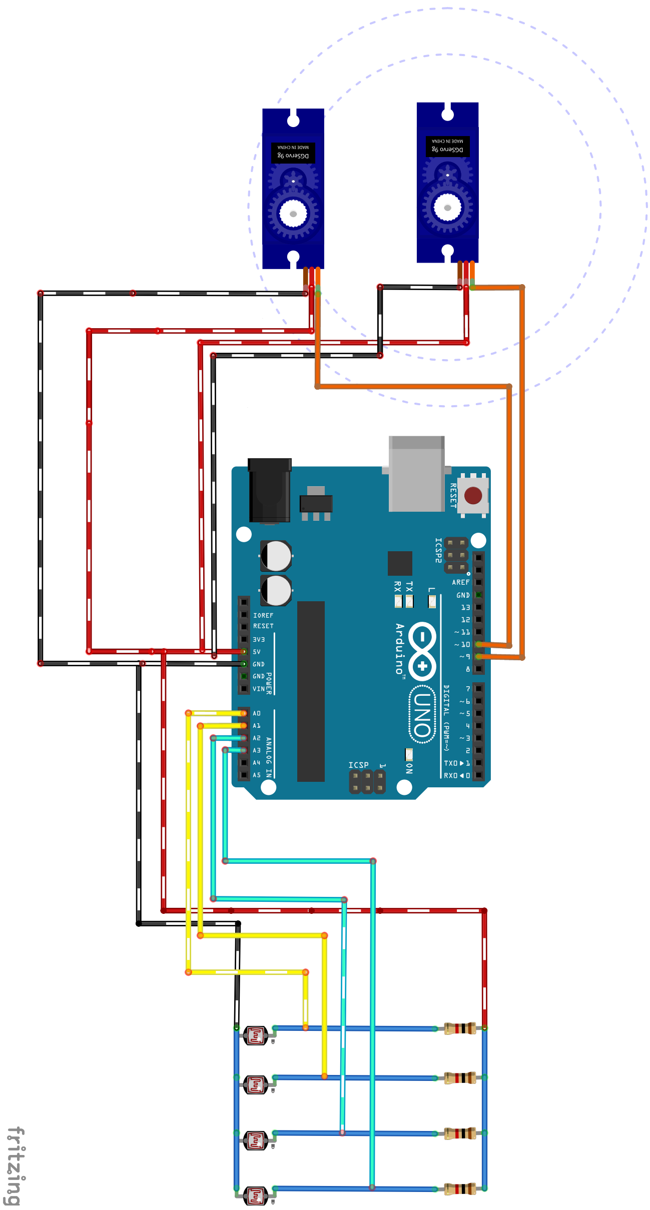
In the prower logger part we also added an additional micro SD module for storing the data in a micro SD card for this portion we have reviewed a youtube video and the link is **https://www.youtube.com/watch?v=5Dp-XatLySM** for this project we also reviewed some articles on MPPT (Maximum Power Point Tracking) and serial and Parallel connection of solar panel although we did not used any of them in this project.

DESIGN:

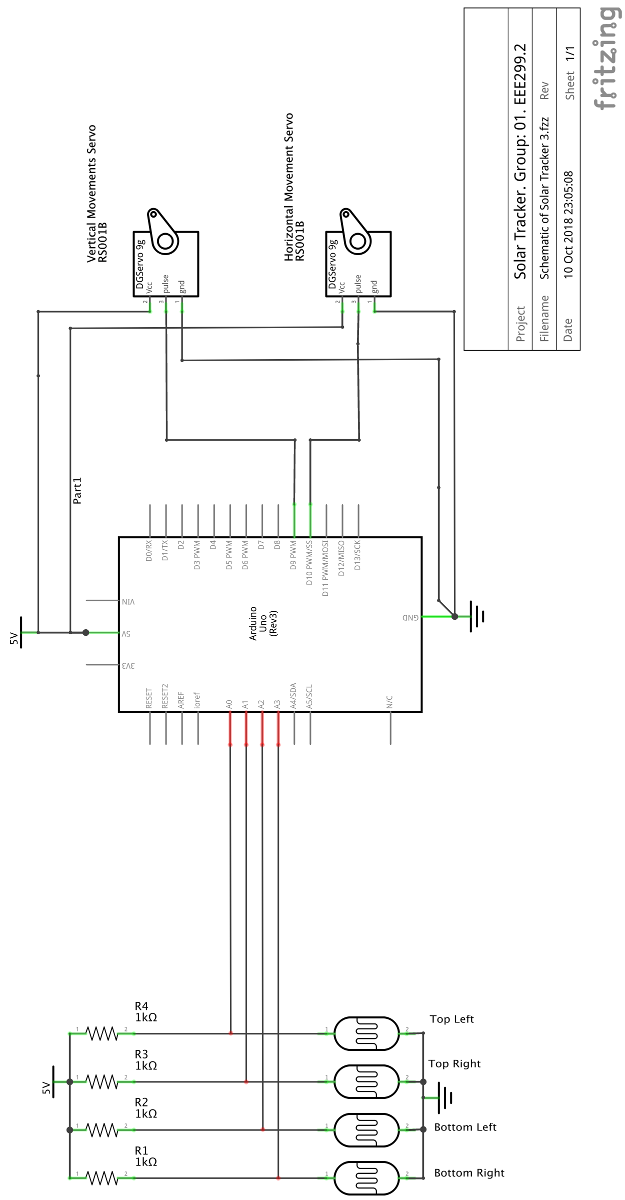
For designing both the breadboard design and schematics we have used Fritzing which is a circuit designing software but it does not have any kind of simulation properties.

SOLAR TRACKER MODULE:

BREADBOARD DESIGN

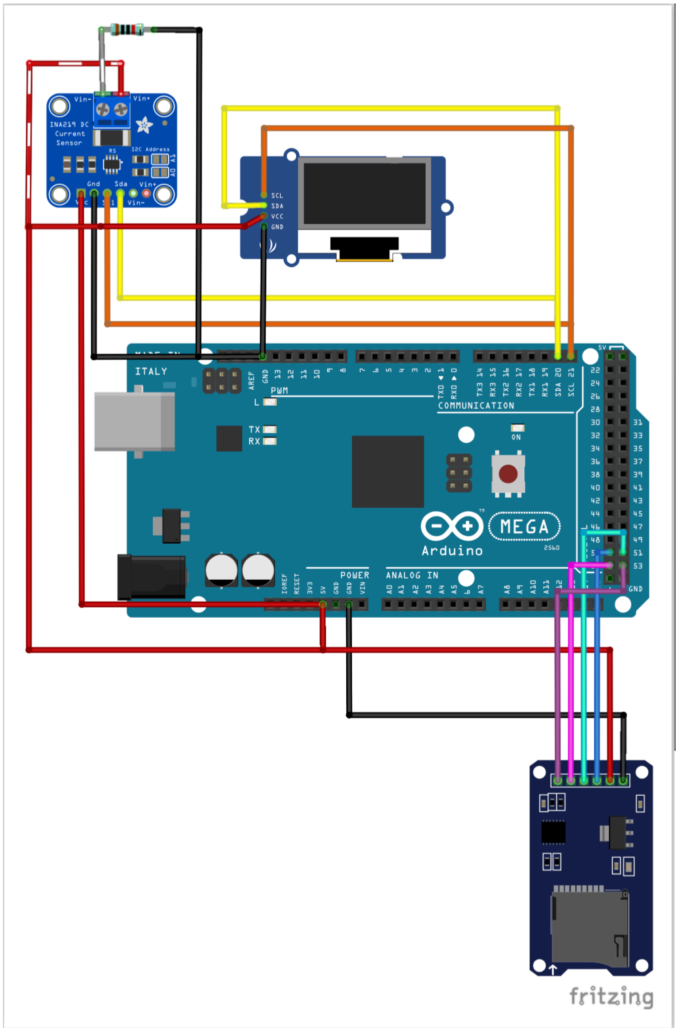


SCHEMATICS

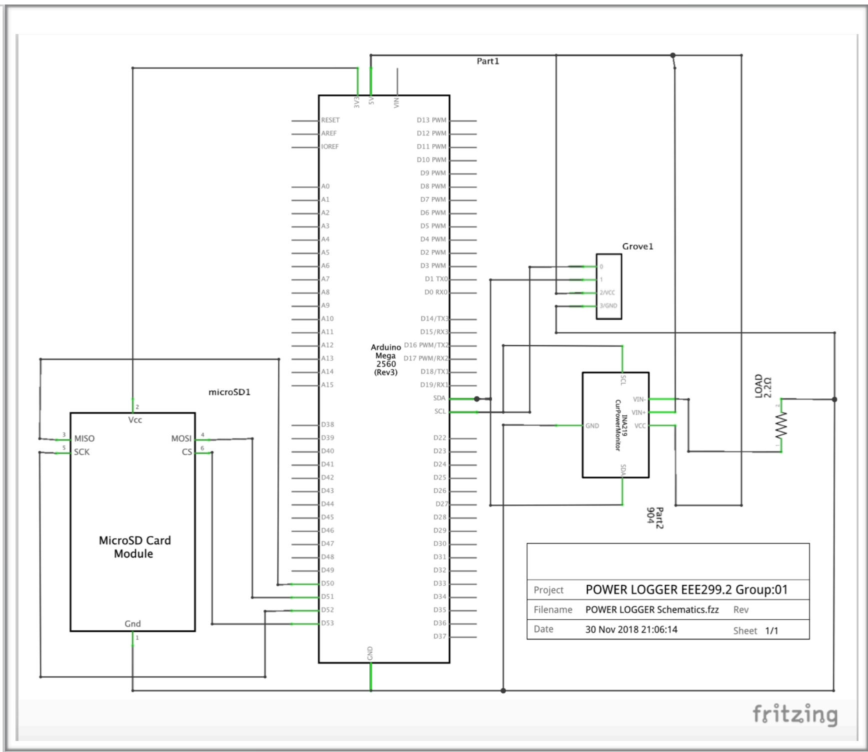


POWER LOGGER MODULE

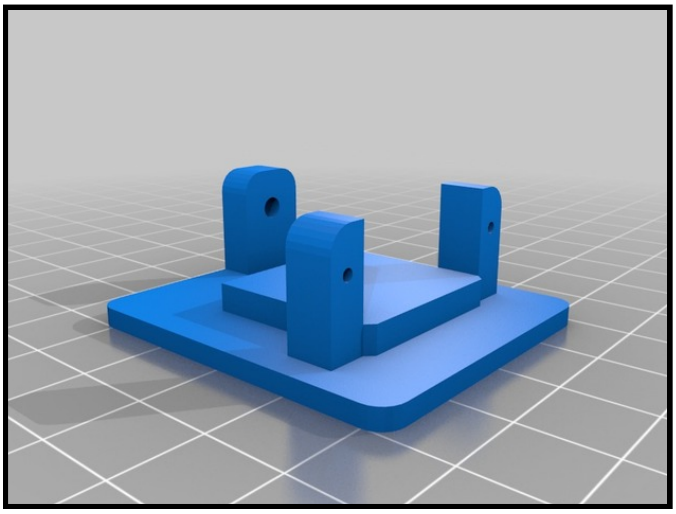
BREADBOARD DESIGN



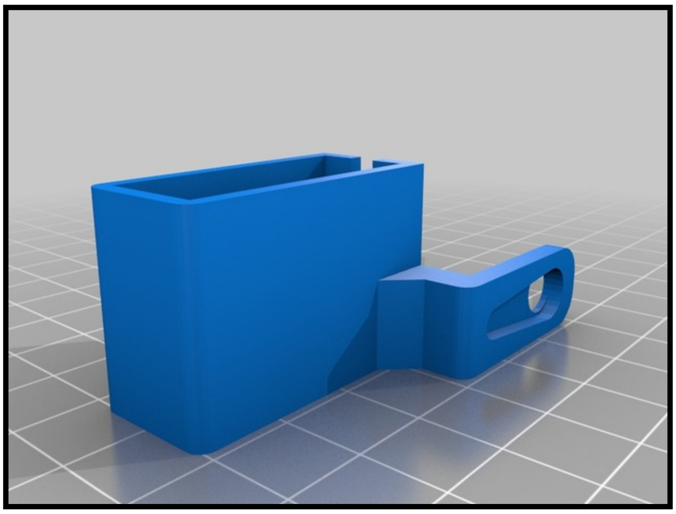
SCHEMATICS



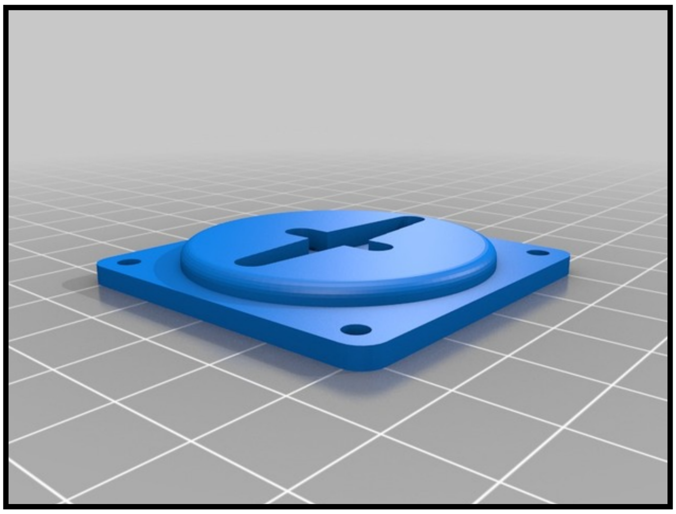
STRUCTURAL DESIGN FOR 3D PRINTING: For the structural design we have used the design made by Fbuenonet. Which can be found on the following web link **https://www.thingiverse.com/thing:708819**



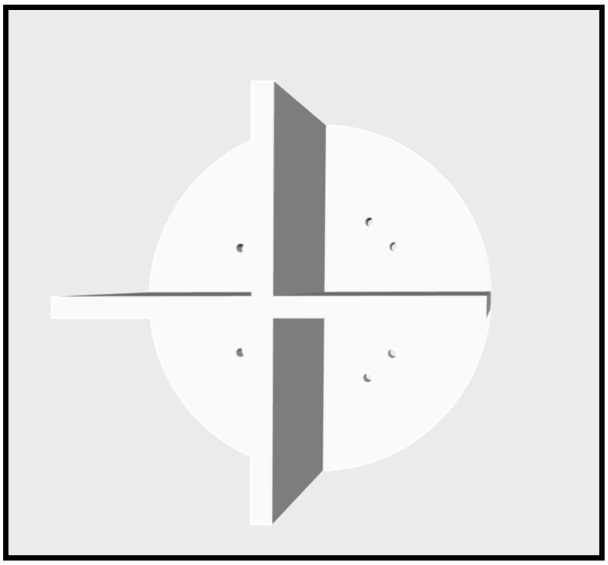
Pantilt Head Part 1



Pantilt Head Part 2



Pantilt Base



LDR Panel

IMPLEMENTATION:

The project has basically two parts:

1. **ROTATING SOLAR PANEL MODULE/SOLAR TRACKER MODULE.**

2. **POWER LOGGER MODULE.**

SOLAR TRACKER MODULE: For this part we have used an Arduino Uno as the brain of the module which actually processes all the data sent by the LDR panel mounted just above the Solar panel. Form this data the Arduino processes the information using a set of codes previously written for the development board. Then the processed information goes to the two servos responsible for the yaw and pitch of the solar panel. For generating the Arduino signal we have utilised the some of the in built registers and comparators of Arduino micro-controller ATmega 328/P. The name of the used registers and comparators are given below:

TC=Timer/Counter
TCCR1A = TC1 Control Register A (page 170 in the data sheet)
TCCR1B = TC1 Control Register B (page 173 in the data sheet)

WGM = Waveform Generation Mode (page 171 in the data sheet)
CTC = Clear Timer on Compare Match (CTC) Mode

COM1 = Compare Output Mode for Channel (page 170 in the data sheet)

ICR1L =Input Capture Register 1 Low byte (page 178 in the data sheet)

ICR1H = Input Capture Register 1 High byte (page 179 in the data sheet)

OCR1A L = Output Compare Register 1 A Low byte (page 180 in the data sheet)

OCR1A H = Output Compare Register 1 A High byte (page 181 in the data sheet)

OCR1B L = Output Compare Register 1 B Low byte (page 182 in the data sheet)

OCR1B H = Output Compare Register 1 B High byte (page 183 in the data sheet)

Output Compare pins (OC1A and OC1B)

CS10, CS11, CS12: Clock Select

CODE FOR SOLAR TRACKER MODULE:

```
int topleft;

int topright;

int downleft;

int downright;

int waittime = 1;


void setup() {

  pinMode(9, OUTPUT); //For Horivontal servo movement

  pinMode(10, OUTPUT); //For Vertical Servo movement

  TCCR1A = 0;

  TCCR1A = (1 << COM1A1) | (1 << COM1B1) | (1 << WGM11);

  TCCR1B = 0;

  TCCR1B = (1 << WGM13) | (1 << WGM12) | (1 << CS11);

  ICR1 = 40000;

  OCR1A = 3000;

  OCR1B = 3600;

}


//LDR input pins

void loop() {

  topleft = analogRead(A0);

  topright = analogRead(A1);

  downleft = analogRead(A2);

  downright = analogRead(A3);
```

```
if (topleft > topright) {     //For Horizontal Movement Start
  OCR1A = OCR1A + 1;
  delay(waittime);
}
if (downleft > downright) {
  OCR1A = OCR1A + 1;
  delay(waittime);
}
if (topleft < topright) {
  OCR1A = OCR1A - 1;
  delay(waittime);
}
if (downleft < downright) {
  OCR1A = OCR1A - 1;
  delay(waittime);
}                              //For Horizontal Movement End
if (OCR1A > 4000) {
  OCR1A = 4000;
}
if (OCR1A < 2000) {
  OCR1A = 2000;
}
if (topleft > downleft) {    //For Vertical Movement start
  OCR1B = OCR1B - 1;
  delay(waittime);
}
```

```
  if (topright > downright) {
    OCR1B = OCR1B - 1;
    delay(waittime);
  }
  if (topleft < downleft) {
    OCR1B = OCR1B + 1;
    delay(waittime);
  }
  if (topright < downright) {
    OCR1B = OCR1B + 1;
    delay(waittime);
  }
  if (OCR1B > 4200) {
    OCR1B = 4200;
  }
  if (OCR1B < 3000) {
    OCR1B = 3000;
  }                              //For vertical Movement end
}
```

POWER LOGGER MODULE: For this part we have used an Arduino mega as the brain of the module and INA 219 IC as the heart. Here the INA 219 IC's Vin+ pin is connected with the source's positive terminal. The load's one portion is connected with Vin- pin and other portion is connected with the negative terminal of the source. From the SCL and SDA pin connected with the Arduino's SCL(pin 21) and SDA(pin 20) pin the INA 219 IC shares it's captured data about the voltage and the current measured by it. Then the Arduino processes those data and shows the data on the OLED screen connected with it using the same SCL(pin 21) and SDA(pin 20) pin. Here we have also connected a Micro SD Storage Board TF Card Shield Module for Arduino to store this data in to a Micro SD card. For tis we have connected the modules MOSI to pin 51, MISO to pin 50, SCK to pin 52 and CS to pin 53 of Arduino. From this captured data we can plot Voltage ,Current and Power graph.

Elaboration of the used pins:

- MISO (Master In Slave Out) - The Slave line for sending data to the master,
- MOSI (Master Out Slave In) - The Master line for sending data to the peripherals,
- SCK (Serial Clock) - The clock pulses which synchronize data transmission generated by the master and one line specific for every device:
- SS (Slave Select) - the pin on each device that the master can use to enable and disable specific devices.
- CS(Chip Select)
- SDA(Data Line)
- SCL(Clock Line)

CODE FOR POWER LOGGER MODULE:

```
#include <Wire.h>

#include <Adafruit_INA219.h>

#include <Adafruit_SSD1306.h>

#include <SPI.h>

#include "SdFat.h"   //Added library for SD card

SdFat SD;
```

```
}

#define OLED_RESET 4
Adafruit_SSD1306 display(OLED_RESET);
Adafruit_INA219 ina219;


unsigned long previousMillis = 0;
unsigned long interval = 1000;   //Taking measurement after 1000 milliseconds or 1 second interval
const int chipSelect = 53;     //CS pin of arduino mega. It differs with board type
float shuntvoltage = 0;
float busvoltage = 0;
float current_mA = 0;
float loadvoltage = 0;
float energy = 0;
File TimeFile;
File VoltFile;
File CurFile;


void setup() {
  SD.begin(chipSelect); //For initiating the data storing process in the SD card
  display.begin(SSD1306_SWITCHCAPVCC, 0x3C);
  //For initiating and displaying the specs on the OLED :SSD1306_SWITCHCAPVCC
  ina219.begin(); //For initiating INA219 bi-directional current sensing ic
}

void loop() {
```

```
 unsigned long currentMillis = millis(); //Taking the millisecond time fro the arduino's clock
 if (currentMillis - previousMillis >= interval)
 {
  previousMillis = currentMillis;
  ina219values();  //Calling the function for reading the values from INA219 bi-directional current 
sensing ic


  TimeFile = SD.open("TIME.txt", FILE_WRITE);
  if (TimeFile) {
   TimeFile.println(currentMillis);
   TimeFile.close();
  }


  VoltFile = SD.open("VOLT.txt", FILE_WRITE);
  if (VoltFile) {
   VoltFile.println(loadvoltage);
   VoltFile.close();
  }


  CurFile = SD.open("CUR.txt", FILE_WRITE);
  if (CurFile) {
   CurFile.println(current_mA);
   CurFile.close();
  }
  displaydata();


//For showing the data in the arduino's serial plotter.
```

```
  Serial.begin(9600);  //initiating the serial plotter
     Serial.print("Volt(V):");
     Serial.println(loadvoltage);
     Serial.print("Current(mA):");
     Serial.println(current_mA);
     Serial.print("Time(msec):");
     Serial.println(currentMillis);
     Serial.print("Power(mW):");
     Serial.println(loadvoltage*current_mA);
 }

}

//Function for displaying the data on the OLED
void displaydata() {
 display.clearDisplay();
 display.setTextColor(WHITE);
 display.setTextSize(1);
 display.setCursor(0, 0); //Setting the position
 display.println(loadvoltage);
 display.setCursor(35, 0);
 display.println("V");
 display.setCursor(50, 0);
 display.println(current_mA);
 display.setCursor(95, 0);
 display.println("mA");
 display.setCursor(0, 10);
```

```
  display.println(loadvoltage * current_mA);
  display.setCursor(35, 10);
  display.println("mW");
  display.setCursor(0, 20);
  display.println(energy);
  display.setCursor(35, 20);
  display.println("mWh ::Group01");
  display.setCursor(58, 10);
  display.println("::EEE299.2");
  display.display();
}

//Function For reading the values of INA219 bidiectional current sensing ic
void ina219values() {
  shuntvoltage = ina219.getShuntVoltage_mV();
  busvoltage = ina219.getBusVoltage_V();
  current_mA = ina219.getCurrent_mA();
  loadvoltage = busvoltage + (shuntvoltage / 1000);
  energy = energy + loadvoltage * current_mA / 3600;
}
```

WORKING PRINCIPLES:

The working principles of the solar tracker module is pretty easy and straight forward. The main sensor that controls the overall rotation of the solar panel is the LDR panel consisting of four LDRs.Two at the top and two at the bottom.

Now if the top left and bottom left LDRs or simply the left side of the LDR panel is dark then the servo motor for horizontal movement moves in a clockwise direction until all the LDRs get the same amount of light . For this the code is:

```
if (topleft > topright) {
  OCR1A = OCR1A + 1;
  delay(waittime);
 }
 if (downleft > downright) {
  OCR1A = OCR1A + 1;
  delay(waittime);}
```

If the top right and bottom right LDRs or simply the right side of the LDR panel is dark then the servo motor for horizontal movement moves in a counter clockwise direction until all the LDRs get the same amount of light. For this the code is:

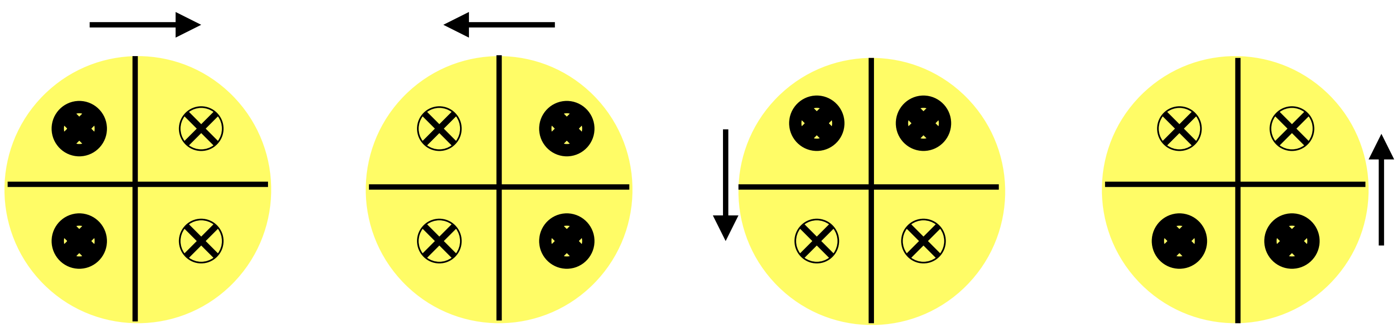
```
if (topleft < topright) {
  OCR1A = OCR1A - 1;
  delay(waittime);
 }
 if (downleft < downright) {
  OCR1A = OCR1A - 1;
  delay(waittime);     }
```

If the top left and top right LDRs or simply the top side of the LDR panel is dark then servo motor for vertical movement moves in the downward direction until all the LDRs get the same amount of light. For this the code is:

```
 if (topleft > downleft) {
   OCR1B = OCR1B - 1;
   delay(waittime);
  }
  if (topright > downright) {
   OCR1B = OCR1B - 1;
   delay(waittime);}
```

If the bottom left and bottom right LDRs or simply the top side of the LDR panel is dark then servo motor for vertical movement moves in the upward direction until all the LDRs get the same amount of light. For this the code is:

```
 if (topleft < downleft) {
   OCR1B = OCR1B + 1;
   delay(waittime);
  }
  if (topright < downright) {
   OCR1B = OCR1B + 1;
   delay(waittime);}
```



Clockwise Direction **Anti Clockwise Direction** **Downward Direction** **Upward Direction**

Now for the question of what degree the servos need to rotate is also quiet simple. From the connection setup we have seen that all the LDRs are connected to Arduino Uno's analog pins A0 to A3 pins and servos are connected with the digital pins D9 and D10. As we have used LDRs which has a maximum 5kΩ resistance in dark and minimum 0Ω resistance (Theoretically) in light. Depending on the intensity of the light the LDR's produce a resistance and that resistance value is given to the analog pins of the Arduino. And using the in built ADC (Analog to Digital Converter) of the Arduino Uno board the analog signal of the LDRs outputs are converted in to a 0 to 5V which are then send to the digital pins which are then fed to the servo's input pins, then the servos again converts the digital voltage signal to analog signals for which the servos rotate to a particular degree of angle. That is how the solar tracker module works.

For the power logger module the main work is done by the INA219 Bi-directional current sensing module. For sensing the current and voltage the module uses the following codes:

```
void ina219values() {
  shuntvoltage = ina219.getShuntVoltage_mV();
  busvoltage = ina219.getBusVoltage_V();
  current_mA = ina219.getCurrent_mA();
  loadvoltage = busvoltage + (shuntvoltage / 1000);
  energy = energy + loadvoltage * current_mA / 3600;
}
```

Costing:

The costing of the total project is illustrated in the table below:

SERIAL NO	ITEM NAME	NUMBER OF UNITS	PER UNIT PRICE	COST (IN TAKA)
1	ARDUINO MEGA	1	850	850
2	ARDUINO UNO	1	450	450
3	OLED DISPLAY	1	500	500
4	INA 219 IC MODULE	1	1380	1380
5	SERVO 9g	2	130	260
6	SOLAR PANEL	1	250	250
7	LDR(5KΩ)	4	10	40
8	RESISTOR(1KΩ)	2	0.5	1
9	PLASTIC WOOD BOARD	1	180	180
10	JUMPER WIRES	2 (SET)	60	120
11	MICRO SD MODULE	1	250	250
12	SD CARD(4GB)	1	250	250
13	BREAD BOARD	1	100	100
14	3D PRINTING COST	1 (SET)	450	450
	TOTAL COST			**4911**

DISCUSSION OF RESULTS:

From the power logger's logged data which was stored in the CUR.txt, VOLT.txt, TIME.txt files in the SD card the following data table is generated for the Solar Tracker or rotating solar panel and also for the flat mounted solar panel. The following experiment was led from 10AM to 1PM on 19/12/2018. Which was in the Winter season thus the efficiency of the solar panels are already low due to low Sunlight intensity. (Here the load used is 2kΩ for both setup & both of the solar panels have the same characteristics)

Data Table for Rotating Solar Panel: (Hourly Average)

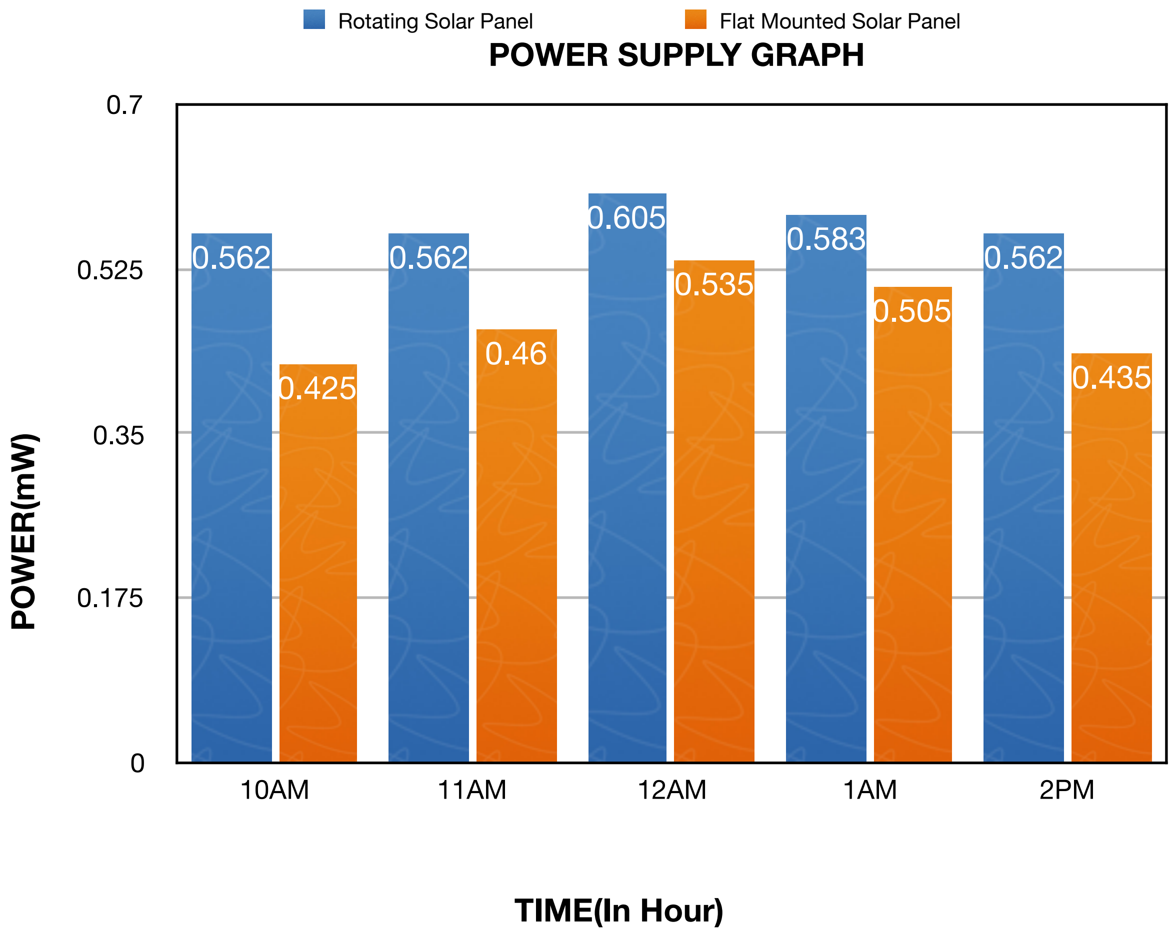
HOUR	VOLTAGE	CURRENT(mA)	POWER(mW)
1	1.06	0.53	0.5618
2	1.06	0.53	0.5618
3	1.10	0.55	0.6050
4	1.08	0.54	0.5832
5	1.06	0.53	0.5618
TOTAL POWER CONSUMED			**2.8736**

Data Table for Flat Mounted Solar Panel: (Hourly Average)

HOUR	VOLTAGE	CURRENT(mA)	POWER(mW)
1	0.85	0.53	0.4250
2	0.92	0.53	0.4600
3	1.07	0.55	0.5350
4	1.01	0.54	0.5050
5	0.87	0.53	0.4350
TOTAL POWER CONSUMED			**2.3600**

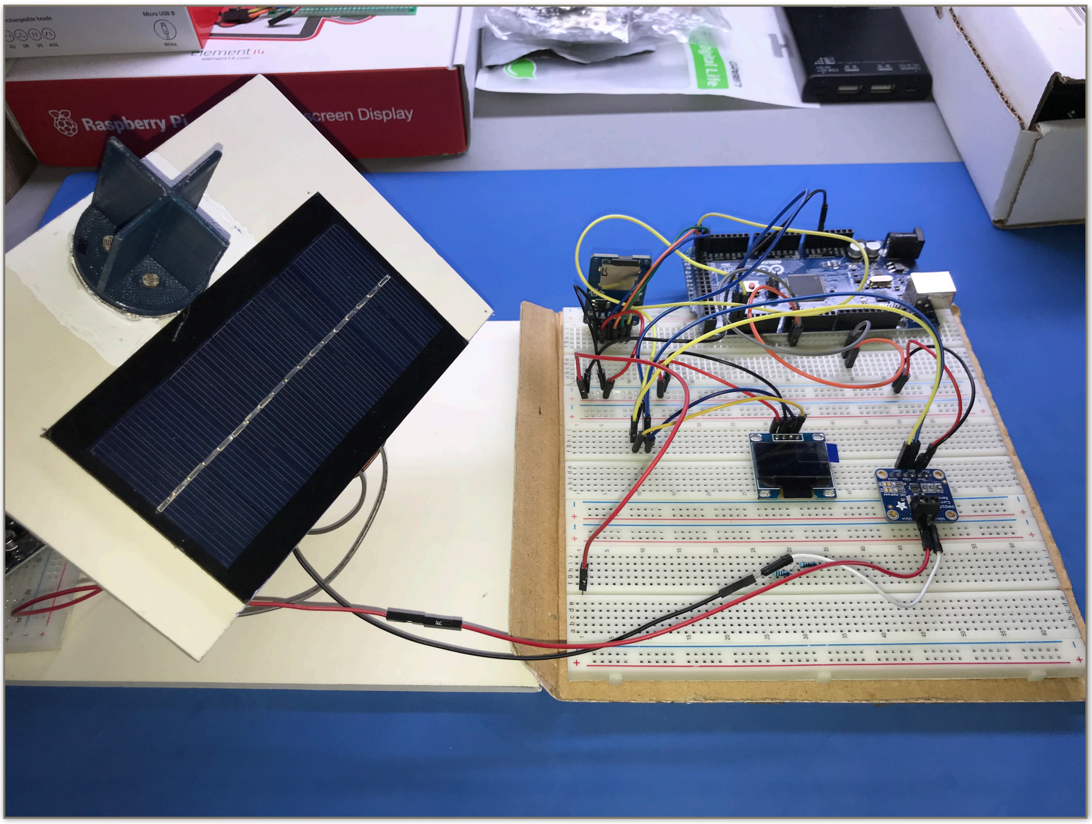
Difference between power supplied = (2.8736 - 2.3600) mW = 0.5136 mW

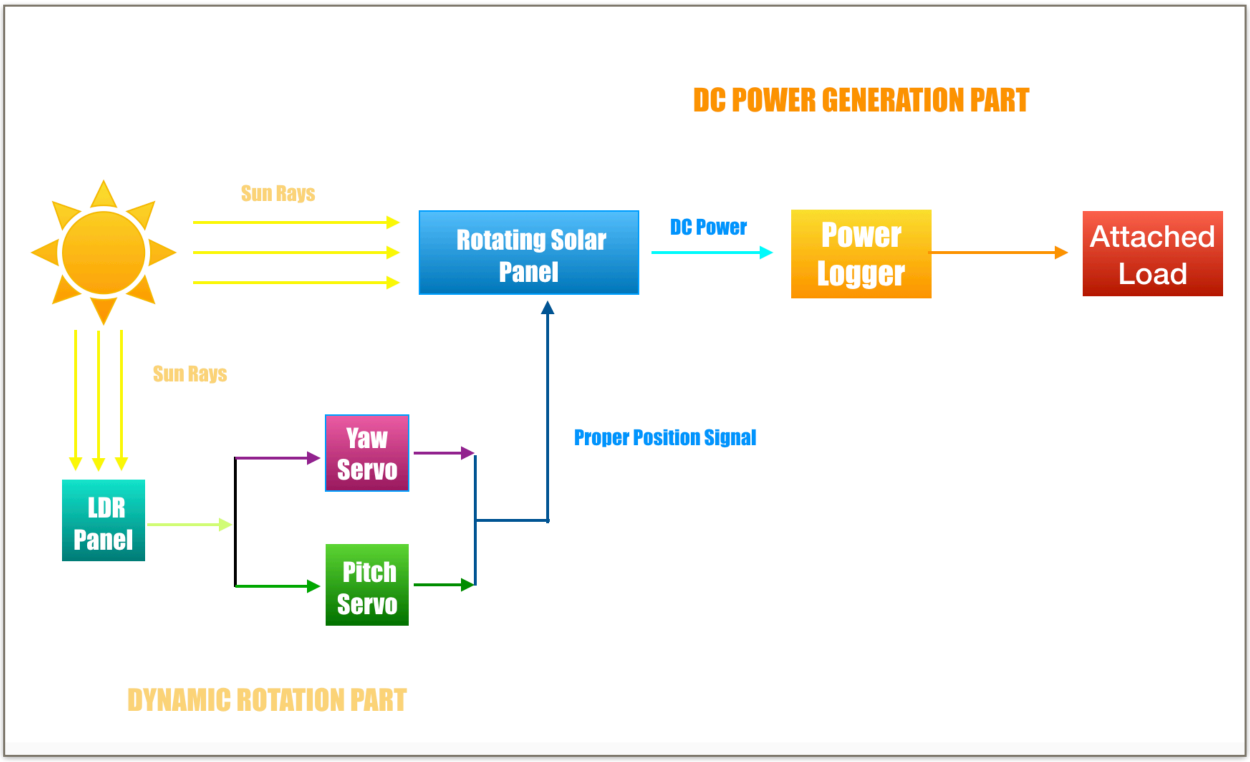
Which is about 21% more than that of a flat mounted solar panel.



Overall Project View:

Original View





Overall view (Theoretical):

APPENDIX:

A

ADC: Analog to Digital Converter

C

CTC :Clear Timer on Compare Match (CTC) Mode

COM1:Compare Output Mode for Channel (page 170 in the data sheet)

CS10, CS11, CS12: Clock Select

CS :Chip Select

I

ICR1L =Input Capture Register 1 Low byte

ICR1H = Input Capture Register 1 High byte

MISO (Master In Slave Out) - The Slave line for sending data to the master,

MOSI (Master Out Slave In) - The Master line for sending data to the peripherals,

O

OCR1A L = Output Compare Register 1 A Low byte

OCR1A H = Output Compare Register 1 A High byte

OCR1B L = Output Compare Register 1 B Low byte

OCR1B H = Output Compare Register 1 B High byte

Output Compare pins (OC1A and OC1B)

S

SCK (Serial Clock) - The clock pulses which synchronize data transmission generated by the master and one line specific for every device:

SS (Slave Select) - the pin on each device that the master can use to enable and disable specific devices.

SDA: Serial Data Line

SCL: Serial Clock Line

T

TC=Timer/Counter

TCCR1A = TC1 Control Register A

TCCR1B = TC1 Control Register B

W

WGM = Waveform Generation Mode

REFERENCES:

1.https://www.arduino.cc

2.https://www.solarreviews.com/blog/what-are-the-most-efficient-solar-panels-for-2018

3.https://www.eia.gov/todayinenergy/detail.php?id=18871

4.https://www.thingiverse.com/thing:708819

5.https://www.youtube.com/watch?v=kUHmYKWwuWs&list=WL&index=43&t=0s

6.https://www.youtube.com/watch?v=BtLwoNJ6klE&list=WL&index=45

7.https://www.youtube.com/watch?v=Caw5PiIFk-w&list=WL&index=51

8.https://www.youtube.com/watch?v=8MvRRNYxy9c&index=52&list=WL

9.https://www.youtube.com/watch?v=5Dp-XatLySM&index=53&list=WL

10.https://www.youtube.com/watch?v=2kr5A350H7E&index=40&list=WL

11.https://www.youtube.com/watch?v=J61_PKyWjxU&index=41&list=WL

12.https://dronebotworkshop.com/lcd-displays-arduino/

13.https://www.youtube.com/watch?v=_e_0HJY0uIo&list=WL&index=36

14.https://cdn-learn.adafruit.com/downloads/pdf/adafruit-ina219-current-sensor-breakout.pdf

15.https://www.sparkfun.com/datasheets/Components/SMD/ATMega328.pdf

16.http://nevonprojects.com/rotating-solar-panel-using-arduino/

17.https://circuitdigest.com/microcontroller-projects/arduino-solar-panel-tracker

18.https://www.instructables.com/id/Simple-Dual-Axis-Solar-Tracker/

19.https://data.worldbank.org/indicator/EG.USE.ELEC.KH.PC