SUN FOLLOWER

INTRODUCTION:

This project helps solar panels capture the maximum amount of solar radiation. In a typical solar field a fixed solar panel only receives about **32%** of the optimum amount of solar radiation because of beam spreading. A solar panel perpendicular to sun at all times, is considered to be at an optimum angle to receive the maximum solar radiation. From this idea, I realized that if I can create a fixture that follows the sun like a ***Sunflower***, we can capture maximum solar radiation. Depending on the day of the year, Sun appears at different locations on the horizon at sunrise, and traces an arc in the sky through sunset. This arc repositions itself throughout the year. My project takes these considerations and manages the positioning of solar panel very closely.

Movements of the project depends on Sun-Earth Geometry, and latitude & longitude of the current location. My thought process is to (1) Figure out the location using GPS Module, thus getting - Latitude, Longitude, UTC Date and Time for the location (2) Using GPS inputs, calculate - (i) Nth Day of the year (ii) Declination of Earth (iii) Hour Angle (iv) Solar Time (v) Equation of Time (vi) Altitude and (vii) Azimuth. Altitude is where Sun appears (or) Sun's position in the Sky at any given time. Azimuth on the other hand is the sweep (angle) of Sun from sunrise to sunset as measured from ***true*** North.

An Arduino Mega microcontroller, manages the logic. Adafruit GPS v3 unit, captures NMEA GPS phrases, from here we proceed to calculate - Altitude and Azimuth Angles for the given date, for the given time and given location. A NEMA 17 motor is used to control the pitch of the Solar Panel and serves as our Altitude motor, while a NEMA 23 motor is used to control the swing (throw) and serves as our Azimuth motor. In the REAL MODE - after obtaining the GPS Fix, we proceed to the calculate true solar time, Altitude and Azimuth angles and move the motors to the appropriate positions. This process repeats every 10 minutes from sunrise until sunset. In the DEMO RUN MODE - we run the clock from 8AM local solar time to 9PM local solar time, and fast track through 1 hr intervals every 20 seconds. In the DEMO SET MODE - User is given a chance to adjust the potentiometers to select from 12 different locations around the earth from Tokyo to Honolulu, and different months of the year and 28 different days.

NDay function calculates the Nth Day for the year. January 1st is the first day of the year and Dec 31 in a leap year is 366th day. This is needed in order to find the position of the earth going around the Sun, like on Jan 2nd, Earth is at the Perihelion, the closest point in its orbit around the Sun at 1.47x1011M. NDay is used in calculation of Solar Time.

Hour Angle function calculates the angular distance between the meridian of the observer and the meridian whose plane has the sun. This can be measured by the formula:

$$w = 15(ts - 12) \quad \text{[ts is the Solar Time]}$$

EOT(Equation of Time) is also known as the difference between the mean solar time and the true solar time. True solar time is assumed to be 12 noon every day but the mean solar time is when then the sun is directly overhead and it ranges from 17 minutes ahead or behind the true solar noon. The equation for this is :

$$EOT = 0.258\cos(x) - 7.146\sin(x) - 3.648\cos(2x) - 9.288\sin(2x)$$

$$\text{where } x = 360(N - 1)/365.242.$$

Declination angle is the the latitude angle in which the sun shines directly overhead. “If a line is drawn between the center of the earth and the sun (ecliptic plane), the angle between this line and the earth's equatorial plane is called the *declination angle”*.

\] sin = 0.39795cos[0.98563(N - 173)] [is the declination angle]

Altitude angle and Azimuth angle depend on declination angle, latitude, hour angle. These formulas are derived from sun pointing vector at the surface of the earth and then mathematically translating them to the center of the earth with a slightly different coordinate system. These Angle serve as inputs to turn the motors in a specific direction. Simply put the altitude is how high the sun is in the sky and the azimuth is the angle of the sun from true north.

ALT = sin-1(sin()*sin(latitude) + cos()*cos(w)*cos(latitude))
AZI = cos-1(sin()*cos(latitude) - cos()*cos(w)*sin(latitude)) / cos(ALT))

OPERATION:

On the control panel, an Adafruit OLED is used for display, two SPDT (single pole, double throw) switches are used to select between Real and Demo Modes and Demo Run and Demo Set Modes. Three potentiometers are used for selection of Place, Month and Day and Reset Button - restarts the program.

In the Real mode the program first tries to acquire GPS fix, if a GPS fix is found it figures out Altitude and Azimuth angles and moves the motors appropriately, it then proceeds to sleep for 10 minutes. In the event GPS fix is not acquired, it defaults to GPS coordinates of Otto Middle School [*Latitude: 33.005803, Longitude: -96.6528103*] for Nov 14th, 2017 and proceeds to run the program in a demo-mode every 20 seconds.

In the Demo mode users have a choice between SET and RUN. In the SET mode, users are given an opportunity to select Location, Day and Month by turning the potentiometers, these variables will serve as inputs for the Demo RUN Mode where upon reading the potentiometer values - the program starts executing a Demo Run from 8AM to 9PM Solar Time for the location incrementing every 20 seconds as 1 hour.

SUN FOLLOWER

INTRODUCTION:

This project helps solar panels capture the maximum amount of solar radiation. In a typical solar field a fixed solar panel only receives about **32%** of the optimum amount of solar radiation because of beam spreading. A solar panel perpendicular to sun at all times, is considered to be at an optimum angle to receive the maximum solar radiation. From this idea, I realized that if I can create a fixture that follows the sun like a ***Sunflower***, we can capture maximum solar radiation. Depending on the day of the year, Sun appears at different locations on the horizon at sunrise, and traces an arc in the sky through sunset. This arc repositions itself throughout the year. My project takes these considerations and manages the positioning of solar panel very closely.

Movements of the project depends on Sun-Earth Geometry, and latitude & longitude of the current location. My thought process is to (1) Figure out the location using GPS Module, thus getting - Latitude, Longitude, UTC Date and Time for the location (2) Using GPS inputs, calculate - (i) Nth Day of the year (ii) Declination of Earth (iii) Hour Angle (iv) Solar Time (v) Equation of Time (vi) Altitude and (vii) Azimuth. Altitude is where Sun appears (or) Sun's position in the Sky at any given time. Azimuth on the other hand is the sweep (angle) of Sun from sunrise to sunset as measured from ***true*** North.

An Arduino Mega microcontroller, manages the logic. Adafruit GPS v3 unit, captures NMEA GPS phrases, from here we proceed to calculate - Altitude and Azimuth Angles for the given date, for the given time and given location. A NEMA 17 motor is used to control the pitch of the Solar Panel and serves as our Altitude motor, while a NEMA 23 motor is used to control the swing (throw) and serves as our Azimuth motor. In the REAL MODE - after obtaining the GPS Fix, we proceed to the calculate true solar time, Altitude and Azimuth angles and move the motors to the appropriate positions. This process repeats every 10 minutes from sunrise until sunset. In the DEMO RUN MODE - we run the clock from 8AM local solar time to 9PM local solar time, and fast track through 1 hr intervals every 20 seconds. In the DEMO SET MODE - User is given a chance to adjust the potentiometers to select from 12 different locations around the earth from Tokyo to Honolulu, and different months of the year and 28 different days.

NDay function calculates the Nth Day for the year. January 1st is the first day of the year and Dec 31 in a leap year is 366th day. This is needed in order to find the position of the earth going around the Sun, like on Jan 2nd, Earth is at the Perihelion, the closest point in its orbit around the Sun at 1.47x1011M. NDay is used in calculation of Solar Time.

Hour Angle function calculates the angular distance between the meridian of the observer and the meridian whose plane has the sun. This can be measured by the formula:

$$w \quad = 15(ts - 12) \quad \text{[ts is the Solar Time]}$$

EOT(Equation of Time) is also known as the difference between the mean solar time and the true solar time. True solar time is assumed to be 12 noon every day but the mean solar time is when then the sun is directly overhead and it ranges from 17 minutes ahead or behind the true solar noon. The equation for this is :

$$EOT = 0.258\cos(x) - 7.146\sin(x) - 3.648\cos(2x) - 9.288\sin(2x)$$

$$\text{where } x = 360(N - 1)/365.242.$$

Declination angle is the the latitude angle in which the sun shines directly overhead. "If a line is drawn between the center of the earth and the sun (ecliptic plane), the angle between this line and the earth's equatorial plane is called the *declination angle"*.

\] $\sin \ = 0.39795\cos[0.98563(N - 173)] \quad$ [is the declination angle]

Altitude angle and Azimuth angle depend on declination angle, latitude, hour angle. These formulas are derived from sun pointing vector at the surface of the earth and then mathematically translating them to the center of the earth with a slightly different coordinate system. These Angle serve as inputs to turn the motors in a specific direction. Simply put the altitude is how high the sun is in the sky and the azimuth is the angle of the sun from true north.

$$ALT = \sin^{-1}(\sin()*\sin(latitude) + \cos()*\cos(w)*\cos(latitude))$$

$$AZI = \cos^{-1}(\sin()*\cos(latitude) - \cos()*\cos(w)*\sin(latitude)) / \cos(ALT))$$

OPERATION:

On the control panel, an Adafruit OLED is used for display, two SPDT (single pole, double throw) switches are used to select between Real and Demo Modes and Demo Run and Demo Set Modes. Three potentiometers are used for selection of Place, Month and Day and Reset Button - restarts the program.

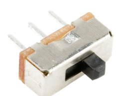
In the Real mode the program first tries to acquire GPS fix, if a GPS fix is found it figures out Altitude and Azimuth angles and moves the motors appropriately, it then proceeds to sleep for 10 minutes. In the event GPS fix is not acquired, it defaults to GPS coordinates of Otto Middle School [*Latitude: 33.005803, Longitude: -96.6528103*] for Nov 14th, 2017 and proceeds to run the program in a demo-mode every 20 seconds.

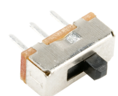
In the Demo mode users have a choice between SET and RUN. In the SET mode, users are given an opportunity to select Location, Day and Month by turning the potentiometers, these variables will serve as inputs for the Demo RUN Mode where upon reading the potentiometer values - the program starts executing a Demo Run from 8AM to 9PM Solar Time for the location incrementing every 20 seconds as 1 hour.




1x NEMA 17 Motor

1x NEMA 23 Motor

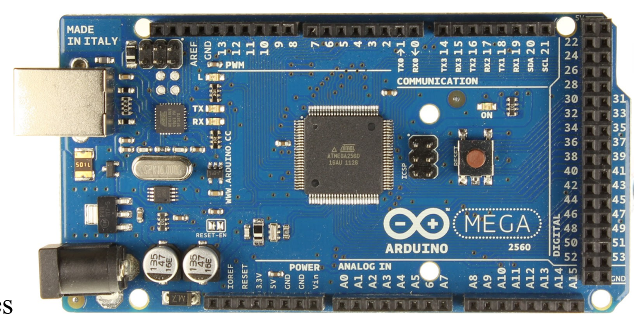





1x wooden base

1x wooden shaft

1x SQ Aluminum shaft



2x Metal Flanges

2x TB6600 Motor drivers

1x 12v-1A Power Supply

1x 128x64 Adafruit OLED

1x Adafruit GPS v3

1x GPS Antenna

1x Arduino Mega

1x Breadboard

2x Momentary Switch w/Rollers

3x Potentiometers

2x SPDT switches

2x Rigid Shaft Couplings

Solder

Male-Female jumper wires

Male-Male jumper wires

CONCLUSION & SUMMARY

This project met its design criteria, the prototype follows the ***Sun*** and collects the maximum amount of solar radiation exposed as stated in my original plan. The azimuth and the altitude angles were compared to the NOAA(National Oceanic and Atmospheric Administration) and the results were very close. [https://www.esrl.noaa.gov/gmd/grad/solcalc/azel.html]

FACTS:

There are many approaches to calculate the Azimuth and Altitude angles, however I used the ones that I could derive, often the simplest for this project. (1) The accuracy of true Solar Time for the location is 16 minutes. This changes the hour angle, which impacts Altitude and Azimuth. (2) The setup is designed to operate from 8AM to 9PM local-solar time, however during the Summer months this may need further adjustment to collect optimal Solar radiation (3) This project is only designed to work in Northern Latitudes. Locations in Southern Hemisphere (eg. South America, Australia, Africa) have equations that orient to true South. (4) For latitudes close to or above the Arctic circle (66° N), there are long Summer Days with 24Hrs of Sun, and this setup will not work. Similarly for winter months near Arctic circle, there is no Sun and this setup will not work. (5) I have used UTC time and longitude to calculate Local Solar Time, which is different from Local Time. Hence this setup does not account for or take into consideration adjustments for Daylight Savings times.

FLAWS:

There are some design flaws in the prototype. (1) One of the main flaws, I put the shaft of the altitude motor right on top of the azimuth motor, which adds unnecessary weight and wobble to the Azimuth motor. In the real world there would be a gearbox driving the shaft, and a bearing to hold the shaft in position. (2) I did not account for Gust and Gale forces, which can swing the entire setup out of control, which could be very dangerous. In the real world, this setup could be achieved with a minimum rise and controlled on 3 axes by Hydraulics (3) I did not account for Magnetic Declination for the location, which can vary as much as 10°. [https://education.usgs.gov/lessons/compass.html]

FUTURE IMPROVEMENTS:

There are many opportunities to improve this design - Developing an iPhone and Android Application which can connect to the setup over BlueTooth and (1) to run the Demo (2) in the event GPS fix cannot be acquired - Apps can provide critical inputs such as latitude and longitude, Date and UTC time (3) Solar Energy data that is gathered at the panels (SD Card) can be uploaded for future improvements.

BIBLIOGRAPHY

1. S. (2011, June 27). Intro to Solar Orientation [Solar Schoolhouse]. Retrieved November 14, 2017, from https://www.youtube.com/watch?v=OR8EQ0DWpPw
2. Earth Sun Geometry. (2012, September 5). Retrieved from https://www.youtube.com/watch?v=rnM1hXJf4WU
3. How do I calculate the Azimuth (angle to north) between two WGS84 coordinates. (n.d.). Retrieved from https://stackoverflow.com/questions/642555/how-do-i-calculate-the-azimuth-angle-to-north-between-two-wgs84-coordinates
4. Power From The Sun :: Chapter 3. (n.d.). Retrieved from http://www.powerfromthesun.net/Book/chapter03/chapter03.htm
5. Declination Angle | PVEducation. (n.d.). Retrieved from http://www.pveducation.org/pvcdrom/properties-of-sunlight/declination-angle

180* tilt of the solar panel.

Solar Panel

Servo Motor

Base