

Trabajo de Fin de Grado
Grado en Ingeniería Electrónica, Robótica y
Mecatrónica

Diseño de un sistema para medir la radiación solar
directa

Autor: Alberto Tomás Muñoz Padilla

Tutor: Eduardo Fernández Camacho

Tutor Externo: Ramón Andrés García Rodríguez

Dpto. Ingeniería de Sistemas y Automática
Escuela Técnica Superior de Ingeniería
Universidad de Sevilla

Sevilla, 2021



Ingeniería de Sistemas y Automática



Trabajo de Fin de Grado
Grado en Ingeniería Electrónica, Robótica y Mecatrónica

Diseño de un sistema para medir la radiación solar directa

Autor:

Alberto Tomás Muñoz Padilla

Tutor:

Eduardo Fernández Camacho

Tutor externo:

Ramón Andrés García Rodríguez

Dpto. de Ingeniería de Sistemas y Automática
Escuela Técnica Superior de Ingeniería
Universidad de Sevilla
Sevilla, 2021

Proyecto Fin de Carrera: Diseño de un sistema para medir la radiación solar directa

Autor: Alberto Tomás Muñoz Padilla
Tutor: Eduardo Fernández Camacho
Tutor externo: Ramón Andrés García Rodríguez

El tribunal nombrado para juzgar el Proyecto arriba indicado, compuesto por los siguientes miembros:

Presidente:

Vocales:

Secretario:

Acuerdan otorgarle la calificación de:

Sevilla, 2021

El Secretario del Tribunal

A mi familia

A mis maestros

Agradecimientos

En primer lugar, me gustaría agradecer a mis padres por su apoyo y esfuerzo durante este tiempo, a mi hermana por servirme de ejemplo.

A mis amigos tanto de Sevilla, de la universidad, así como a los de mi pueblo, Úbeda; por los buenos momentos y risas durante estos años.

La realización de este trabajo tampoco hubiese sido posible de no ser por la ayuda de los tutores durante todo este viaje y también a Juan de Dios Yañez Ávila por su contribución durante el proceso de diseño en 3D y también a Manolo, que también me ayudó y me guió en diversos aspectos del diseño e impresión.

Alberto Tomás Muñoz Padilla

Grado de Ingeniería Electrónica, Robótica y Mecatrónica

Sevilla, 2021

Con este trabajo de fin de grado se intentará diseñar, programar y construir un sensor de radiación solar por accionamiento directo para poder medir la irradiación directa procedente del Sol. Más adelante iremos viendo los requisitos y procedimientos a seguir para el diseño y fabricación de un prototipo.

Luego una vez hemos diseñado e impreso los componentes necesarios, se hará un prototipo, con el cual analizaremos en diferentes experimentos su comportamiento en cielo abierto, analizaremos dichos datos y procederemos a mejorarlos, implementando para ello, diferentes algoritmos o técnicas de control para mejorar su precisión y conseguir un resultado más efectivo y acertado.

Por último, procederemos nuevamente a hacer pruebas y tests para ver la mejora en la medición de la irradiación y poder comparar con los resultados anteriores.

Abstract

With this final degree project, we will try to design, program and build a direct drive solar radiation sensor to measure the direct irradiation coming from the sun. Later we will see the requirements and procedures to follow for the design and manufacture of a prototype.

After, once we have designed and printed the necessary components, we will make a prototype, in which we will analyze in different experiments its behavior in open sky, we will analyze those data and proceed to improve them, introducing different algorithms or control techniques to improve their accuracy and achieve a more effective and accurate result.

Finally, we will proceed again to perform tests and trials to see the improvement in the irradiance measurement and to compare with the previous results.

Índice

Agradecimientos	ix
Resumen	xi
Abstract	xiii
Índice	xiv
Índice de Tablas	xvi
Índice de Figuras	xviii
Notación	xx
1 Introducción	1
1.1. <i>Irradiancia Solar</i>	2
1.2. <i>Componentes de la irradiancia solar</i>	2
1.3. <i>¿Cómo se mide la irradiancia solar?</i>	3
1.4. <i>Centrales termosolares</i>	4
2 Hardware	7
2.1. <i>Brujula HMC5883L</i>	7
2.2. <i>LDR 7516</i>	9
2.3. <i>Arduino Micro</i>	10
2.4. <i>Micro Servos</i>	10
2.5. <i>Pan and tilt</i>	11
2.6. <i>Panel Solar</i>	12
2.7. <i>Convertidor de voltaje LM2596</i>	12
2.8. <i>Ventiladores</i>	13
2.9. <i>Conexiones</i>	13
2.9.1. <i>Fritzing</i>	13
2.9.2. <i>Diseño</i>	13
3 Diseño CAD	13
3.1. <i>FreeCAD</i>	13
3.2. <i>SolidWorks</i>	14
3.3. <i>Cura</i>	14
3.4. <i>Base del pan and tilt</i>	14
3.5. <i>Soporte LDRs</i>	
4 Ecuaciones solares	20
4.1. <i>Introduccion</i>	20
4.2. <i>Azimut y altitud</i>	23
4.2.1. <i>Acimut</i>	23
4.2.2. <i>Altitud</i>	23
4.2.3. <i>Cálculo</i>	24
5 Experimentos	27
5.1. <i>Bucle abierto</i>	27
5.1.1. <i>Medición sin tubo</i>	27
5.1.2. <i>Medición con tubo de 10 cm</i>	29

5.1.3.	Filtro óptico	30
5.1.4.	Medición con tubo de 5 cm y filtro óptico	30
5.1.5.	Medición con tubo de 10 cm y filtro óptico	32
5.1.6.	Medición con tubo de 5 cm y filtro óptico al máximo	33
5.1.7.	Medición con tubo de 10 cm , filtro óptico y tapón	34
5.2.	<i>Algoritmos de mejora</i>	36
5.3.	<i>Experimentos con capas de filtro al máximo y tubo de 5cm</i>	38
5.3.1.	Medición con algoritmo aleatorio	38
5.3.2.	Medición con algoritmo en espiral	40
5.3.3.	Medición con algoritmo basado en medidas de las LDRs	40
5.4.	<i>Experimentos con filtro, tubo de 5cm y tapoón</i>	43
5.4.1.	Medición con algoritmo aleatorio	43
5.4.2.	Medición con algoritmo en espiral	44
5.4.3.	Medición con algoritmo basado en medidas de las LDRS	45
6	Conclusiones	48
6.1.	<i>Conclusiones</i>	48
6.2.	<i>Mejoras a realizar</i>	48
7	Referencias	50
8	Anexos	51

ÍNDICE DE TABLAS

Tabla 1. Características de la Brújula digital HMC5883L	7
Tabla 2. Características de la LDR 7516	8
Tabla 3. Características del Arduino Micro	9
Tabla 4. Características del Micro Servo SG90	10
Tabla 5. Características del Panel Solar de 20W	12
Tabla 6. Características del Convertidor Regulador LM2596	12
Tabla 7. Características del Micro Ventilador	13

ÍNDICE DE FIGURAS

Figura 1. Espectro Solar dentro del Espectro Electromagnético	2
Figura 2. Tipos de irradiaciones solares	3
Figura 3. Sección de un pirheliómetro	4
Figura 4. Central Termosolar de Cilindros Parabólicos	5
Figura 5. Planta termosolar de Heliostatos Solucar PS10	5
Figura 6. Disco parabólico de Stirling	6
Figura 7. Brújula digital HMC5883L	7
Figura 8. Fotorresistencia 7516	8
Figura 9. Recta logarítmica del calibrado de la LDR	9
Figura 10. Arduino Micro	9
Figura 11. Microservo	10
Figura 12. Pan and tilt	11
Figura 13. Panel Solar	11
Figura 14. Regulador LM2596	12
Figura 15. Mini ventilador	13
Figura 16. Conexión de los componentes en Fritzing	14
Figura 17. Placa perforada conectada	14
Figura 18. Distintas vistas del soporte preliminar	17
Figura 19. Diseño segundo de la base preliminar	17
Figura 20. Diseño final de la base	18
Figura 21. Impresión de la base	18
Figura 22. Soporte LDRs y pared extraíble preliminar	19
Figura 23. Encaje de las paredes	19
Figura 24. Perfil e interior de la pieza	20
Figura 25. Segunda versión del diseño del soporte y de la pared preliminar	20
Figura 26. Diseño final de ambas partes	21
Figura 27. Resultado de la impresión	21
Figura 28. Vector S en el sistema de referencia con centro en la Tierra	22
Figura 29. Variación de la declinación	23
Figura 30. Alteración de la ecuación del tiempo	23
Figura 31. Vector E en el sistema de referencia ij	24
Figura 32. Punto representado en el sistema XYZ	24
Figura 33. Ángulo acimut	25
Figura 34. Elevación	25
Figura 35. Vector S con sistema de referencia local	26
Figura 36. Montaje completo del seguidor solar	27

Figura 37. Medidas del 18 de mayo	28
Figura 38. Curva de Irradiación Solar	28
Figura 39. Suma de medidas 18 de mayo	29
Figura 40. Tubos de plástico	30
Figura 41. Filtro óptico	30
Figura 42. Medidas del 2 de junio	31
Figura 43. Suma de medidas 2 de junio	31
Figura 44. Medidas del 4 de junio	32
Figura 45. Suma de medidas 4 de junio	32
Figura 46. Medidas del 8 de junio	33
Figura 47. Suma de medidas 8 de junio	34
Figura 48. Medidas del 7 de junio	35
Figura 49. Suma de medidas 7 de junio	35
Figura 50. Algoritmo basado en puntos aleatorios	37
Figura 51. Algoritmo basado en espiral	37
Figura 52. Algoritmo basado en la medida de las LDRs	38
Figura 53. Medidas del 10 de junio	39
Figura 54. Suma de medidas 10 de junio	39
Figura 55. Medidas del 12 de junio	39
Figura 56. Suma de medidas 12 de junio	40
Figura 57. Medidas 23 de junio	40
Figura 58. Suma de medidas 23 de junio	41
Figura 59. Altitud y acimut 5 de julio	41
Figura 60. Diferencia de valores 5 de julio	42
Figura 61. Medias 13 de junio	42
Figura 62. Suma de medidas 13 de junio	43
Figura 63. Medidas 20 de junio	44
Figura 64. Suma de medidas 20 de junio	45
Figura 65. Medidas 21 de junio	45
Figura 66. Suma de medidas 21 de junio	46
Figura 67. Altitud y acimut 4 de julio	46
Figura 68. Diferencia de valores 4 de julio	47
Figura 69. Attiny1626	49
Figura 70. ESP32	49

Notación

A^*	Conjugado
c.t.p.	En casi todos los puntos
c.q.d.	Como queríamos demostrar
■	Como queríamos demostrar
e.o.c.	En cualquier otro caso
e	número e
Re	Parte real
Im	Parte imaginaria
sen	Función seno
tg	Función tangente
arctg	Función arco tangente
sen	Función seno
$\sin^x y$	Función seno de x elevado a y
$\cos^x y$	Función coseno de x elevado a y
Sa	Función sampling
sgn	Función signo
rect	Función rectángulo
Sinc	Función sinc
$\partial y \partial x$	Derivada parcial de y respecto
x°	Notación de grado, x grados.
$\Pr(A)$	Probabilidad del suceso A
SNR	Signal-to-noise ratio
MSE	Minimum square error
:	Tal que
<	Menor o igual
>	Mayor o igual
\	Backslash
\Leftrightarrow	Si y sólo si

1 INTRODUCCIÓN

Cuando el sol se ha puesto, ninguna vela puede reemplazarlo.

- George R.R. Martin -

La finalidad de este trabajo se basará en el desarrollo y construcción un sensor de radiación solar por accionamiento directo de bajo coste, usando para ello un microcontrolador (Arduino) y varios sensores LDR. En este capítulo hablaremos y explicaremos qué es la irradiancia, cuál es la que mediremos y con qué instrumentos se suele medir.

Este trabajo se encuentra enmarcado dentro del Proyecto Europeo Ocontsolar, Optimal Control of Thermal Solar Energy Systems, y guiado por Ramón Andrés García Rodríguez y por el Catedrático Eduardo Fernández Camacho, responsable del Proyecto Europeo



El Proyecto Europeo OCONTSOLAR pertenece al programa de investigación e innovación, Horizonte 2020, tiene la subvención Advanced Grant, concedida por parte del ERC, European Research Council, de la Comisión Europea.

El objetivo del ERC es respaldar las exploraciones más avanzadas iniciadas por los investigadores en todos los campos de investigación, gracias a un sistema justo y competitivo de financiación, con arreglo al criterio de la excelencia científica

El proyecto tiene como objetivo desarrollar nuevos métodos de control para usar sensores móviles montados en drones y vehículos terrestres no tripulados (UGV) como parte fundamental del control de sistemas. Plantas solares serán usadas para casos de estudio, con el objetivo de optimizar su operación usando predicciones y estimaciones de la irradiancia espacial.

En este capítulo también hablaremos de las distintas clases y tipos de plantas solares para las cuales será de

ayuda este trabajo de fin de grado.

1.1 Irradiancia Solar.

El Sol es una fuente de energética que emite un flujo radiante de $3.85 \cdot 10^{26}$ W, de los cuales $1.1 \cdot 10^{17}$ son absorbidos por el planeta [1] y llegan en forma de ondas electromagnéticas, estas ondas tienen diferentes longitudes haciendo que solo una parte de estas ondas puedan ser percibidas por el ojo humano, al conjunto de las ondas totales se les conoce como espectro solar de las cuales, a parte de las visibles por el hombre (40%), tenemos las infrarrojas (50%) y las ultravioletas (10%) como se puede apreciar en la Figura 1.

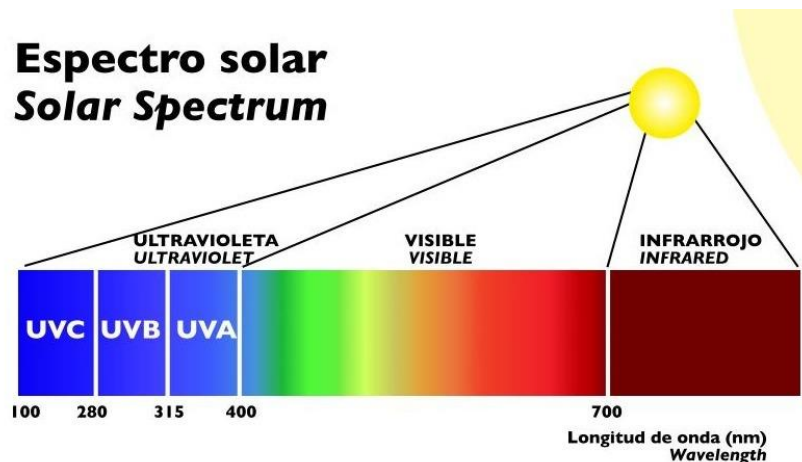


Figura 1. Espectro solar

Como puede verse los rayos ultravioletas se dividen en tres tipos: UVA; atraviesan la atmósfera y alcanza toda la superficie terrestre, los UVB; tienen dificultad para atravesar la atmósfera y los UVC que al tener una longitud de onda tan corta son atrapados por la capa de ozono y no llegan a la superficie.

Luego, la irradiancia solar es la magnitud que nos mide la potencia en una superficie colocada en un lugar y tiempo determinado. La irradiancia se expresa en unidades de potencia por superficie (W/m^2).

1.2 Componentes de la irradiancia solar.

Cuando la radiación solar traspasa la atmósfera, esta se atenúa debido a la absorción por las moléculas de los gases que se encuentran en la atmósfera, en las nubes y las partículas en suspensión. Como consecuencia de esto, se modifica la distribución espectral de la radiación solar, y se reduce la magnitud del flujo radiante, que alcanza la superficie en cuatro formas distintas, directa, difusa, reflejada y total [2], observado en la Figura 2:

- **Irradiancia Normal Directa:** es la que proviene directamente del Sol, sin sufrir cambios de dirección, esta será la que mediremos con nuestro dispositivo.
- **Irradiancia Difusa:** es la que recibimos a través de la atmósfera como consecuencia de la dispersión de parte de la radiación en la propia atmósfera, puede llegar a ser un 15% de la radiación global en días soleados mientras que en nublados este porcentaje es mucho mayor, es destacable comentar que las superficies horizontales son las que más reciben pues están expuestas a toda la semiesfera celeste.
- **Irradiancia Reflejada:** se trata de la radiación reflejada por la superficie terrestre, pero no será siempre la misma cantidad, depende de la superficie y de su coeficiente de reflexión. Aquí al contrario que en la difusa, las superficies horizontales no reciben ninguna radiación al estar de espaldas a la superficie.
- **Irradiancia Total:** es la suma de las 3 irradiaciones anteriores.

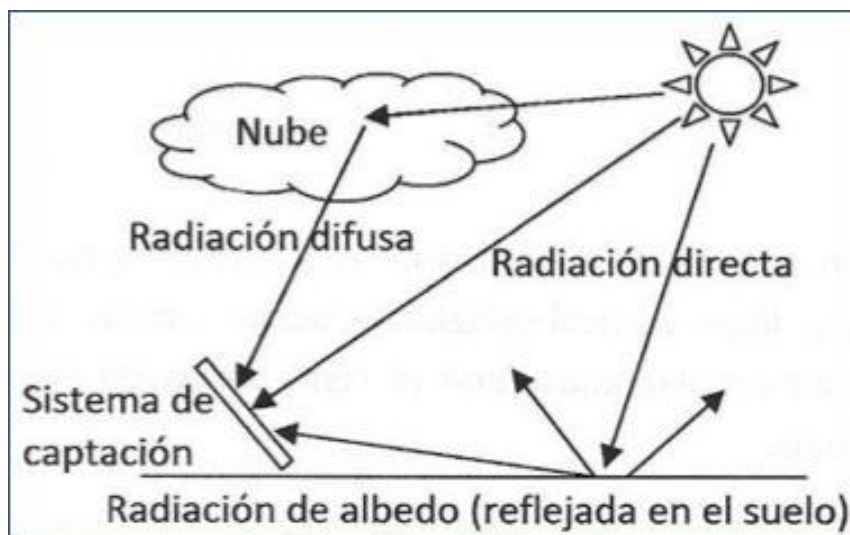


Figura 2. Tipos de irradiaciones

1.3 ¿Cómo se mide la irradiancia solar?

Los dispositivos que permiten esta medida se agrupan en 4 tipos:

1. **Sensores calorimétricos:** la energía radiante es transformada en flujo de calor, que puede ser medido por varios métodos, como el incremento de temperatura de un metal.
2. **Sensores termomecánicos:** el flujo radiante se calcula mediante la magnitud de la flexión que se produce en una lamina formada por dos metales que tienen coeficientes de dilatación térmica bastante distintos.
3. **Sensores termoeléctricos:** partiendo de dos cables de materiales distintos unidos por sus extremos, la irradiación se puede obtener a partir de la fuerza electromotriz originada cuando a uno lo sometemos al flujo radiante mientras que el otro sigue oculto, estos componentes se agrupan formando termopila
4. **Sensores fotoeléctricos:** tienen su principio en la corriente eléctrica que se genera al incidir el flujo en un elemento formado por materiales semiconductores.

La radiación solar se mide mediante unos instrumentos llamados radiómetros. En función del tipo de radiación a medir, usaremos uno u otro. Dado que el trabajo se centrará en la medida de la Irradiancia Normal Directa, se hablará solamente del instrumento necesario.

Para la irradiación directa se mide con un pirheliómetro. Estos deben de seguir el movimiento del Sol, por lo que montan un seguidor solar que lo mantiene siempre alineado con el centro de este. La organización Meteorológica ha establecido 3 categorías para ellos, según diversos factores, como sensibilidad, la estabilidad de los factores de calibración:

- Pirheliómetros absolutos: son aquellos que se autocalibran, evaluándose la irradiación a partir de parámetros físicos del instrumento.
- Pirheliómetros de referencia: son los que se calibran frente al grupo de Patrones Mundiales, en las comparaciones interpirheliométricas una vez cada cinco años.
- Pirheliómetros secundarios: se utilizan de forma operacional en las redes radiométricas, son más sencillos y robustos que los otros dos; se calibran por comparación con los patrones nacionales.

Se basa en un tubo donde uno de sus extremos tenemos una lente para evitar Irradiancias Difusas o Reflejadas y en el otro tenemos un sensor termoeléctrico en donde el flujo de calor se transforma en energía eléctrica, un

esquema del dispositivo se puede ver en la Figura 3. Después mediante diversas expresiones y ecuaciones podemos obtener Irradiancia Normal Directa.

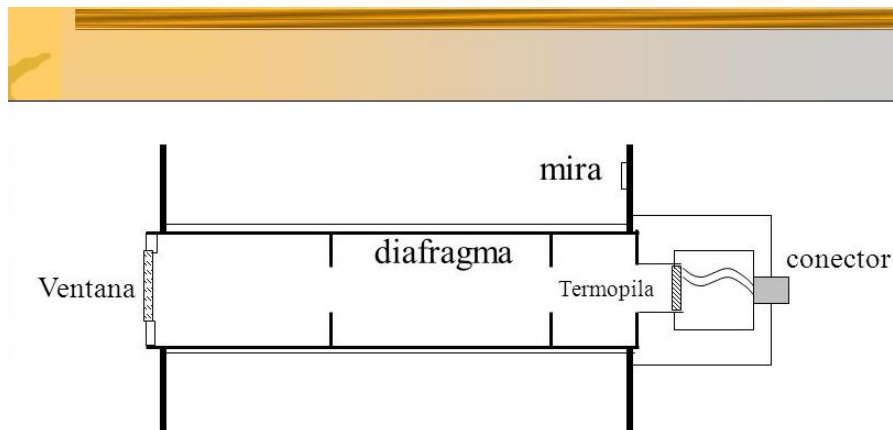


Figura 3. Pirheliómetro

Hay también que tener en cuenta que para la medición de la radiación solar los pirheliómetros deben de cumplir con la norma ISO 9060 [3]. La cual es la norma internacional para la especificación y clasificación de instrumentos de medida de la radiación solar hemisférica y directa, sirve para definir la respuesta a la radiación solar y los límites de las clasificaciones de los pirheliómetros, aunque no para determinar como calcular sus valores, pues hay varias formas de hacerlo.

1.4 Centrales termosolares

Ahora comentaremos la utilidad o para que nos sirve poder captar la Irradiancia Normal Directa. La energía termosolar lleva entre nosotros desde hace mucho tiempo, antiguamente ya se usaban los rayos solares combinados con espejos para encender fuegos y calentar comidas, pero no fue hasta la crisis del petróleo de los años 70, en donde tanto este tipo de energía renovable y otras, empezaron a considerarse como una alternativa a los combustibles fósiles dado el alto precio que habían alcanzado, su investigación y su uso aumentó considerablemente con el paso del tiempo.

Con el avance de la tecnología estas centrales han visto su eficiencia mejorada, sus costes dismuidos y se les considera una verdadera opción en auge frente a otras energías contaminantes, podemos destacar varios tipos de estas [4]:

- **Central termosolar de cilindros parabólicos**

Esta tecnología usa espejos cóncavos montados en un largo eje para dirigir los rayos del Sol hacia el fluido que los atraviesa. Estos espejos son usados para concentrar los rayos solares, son recibidos sobre una tubería por la que circula un fluido, este se transporta a un intercambiador donde cederá su energía térmica para transformarse en energía eléctrica o para almacenarse.

Estos espejos son llamados colectores cilíndricos parabólicos (CCP), estos se orientan al sur (o al norte si están en el hemisferio sur) e incorporan un dispositivo de seguimiento solar el cual los girará para seguir la trayectoria solar a lo largo del día, este dispositivo puede disponer de un eje de actuación, moviéndose en función de la altitud o a través de dos, altitud y acimut. Una imagen de estos vemos en la Figura 4.



Figura 4. Central termosolar CCP

- **Central termosolar de Heliostatos con receptor central en torre**

Los heliostatos son grandes espejos montados sobre una superficie, estos son planos y tienen capacidad de moverse en los dos ejes. Su misión es captar los rayos solares siguiendo la trayectoria descrita por el Sol y reflejar los rayos en una parte concreta de la torre central. En ese punto se suman todos los rayos procedentes de todos los heliostatos y habrá una temperatura muy elevada en donde se calientan unas sales fundidas que tienen gran capacidad calorífica y se usarán para almacenar energía antes de usarla para hervir agua que se convertirá en vapor para mover las turbinas y generar electricidad.

Para esta orientación es preciso un control de las bases de dichos heliostatos, se emplean sistemas que calculan la trayectoria y posicionamiento pues cada heliostato tiene una ubicación distinta luego requiere de un posicionamiento distinto y una regulación particular.

Se puede destacar que la primera que se explotó comercial fue la Solucar PS10, construida por Abengoa en el 2006 con una capacidad instalada de 11 MW y una producción anual de 24GWh, en la Figura 5 podemos ver como es dicha planta.



Figura 5. Planta termosolar de Heliostatos, PS10.

- **Central termosolar con discos parabólicos de Stirling**

Son concentradores independientes. En el foco hay un motor térmico de combustión externa, que transforma la energía recibida por los rayos solares en un giro mecánico. El motor Stirling lleva acoplado un generador eléctrico, luego cada concentrador puede producir energía por sí solo.

Aquí cada concentrador posee un dispositivo de seguimiento solar, el cual lo orienta para poder aprovechar la trayectoria solar, en la Figura 6 se ve un ejemplo de este tipo de concentrador.



Figura 6. Disco parabólico de Stirling

Una vez ya descritos los tipos de plantas termosolares, vemos el importante uso del sensor de radiación por accionamiento directo para poder determinar la trayectoria del Sol y poder monitorizarlo haciendo que los discos, espejos o heliostatos lo sigan y aprovechar al máximo las horas visibles de luz solar para obtener energía.

2 HARDWARE

A continuación, se describirá y detallará los elementos de Hardware que han sido necesarios para la elaboración del trabajo además de las características principales.

2.1 Brújula HMC5883L

Esta brújula digital, es un sensor que nos permite medir el valor del campo magnético en los tres ejes, gracias a eso podemos estimar la orientación del dispositivo respecto al campo magnético de la Tierra. El chip incorpora tres sensores de magnetorresistencia, que le confiere una precisión de $\pm 2^\circ$. La comunicación se realiza a través del Bus I2C, la tensión de voltaje es de 1.8V a 3.3V, aunque incluyen un regulador de voltaje que permite alimentarlos directamente a 5V. Un ejemplo de este dispositivo lo podemos ver en la Figura 7.

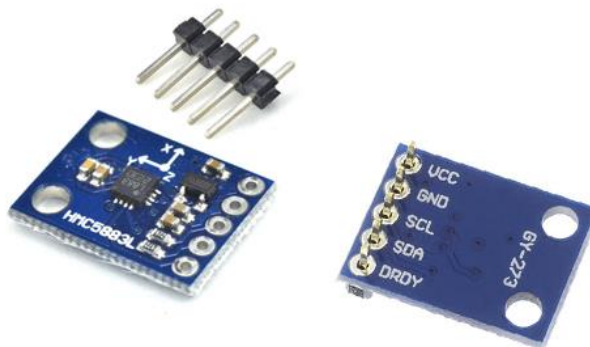


Figura 7. Brújula digital HMC5883L

Voltaje de alimentación	3.3V a 5V
Modelo I2C	GY-271
Chip	HMC5883L
Precisión	$\pm 2^\circ$
Rango de medición	$\pm 1,3 - 8$ Gauss
Resolución	5 mili Gauss
Dimensiones físicas	14 mm x 15 mm

Tabla 1: Características de la Brújula digital HMC5883L

2.2 LDR 7516

Las LDR o fotorresistencias (Light Depending Resistor) son componentes foto electrónicos en los que su resistencia interna varía en función de la luz que incide en él. Al principio y con mucha luz es de unos pocos ohmios y conforme disminuye la luz esta aumentará. Los valores de las resistencias con la luz dependen mucho del tipo de LDR a usar. Gracias a estos sensores nos es posible obtener la cantidad de luz que nos llega del Sol. Una LDR la podemos contemplar en la Figura 8.



Figura 8. Fotorresistencia 7516

Resistencia a 10 Lux	10K a 20K Ohm
Tolerancia a 25°C	500mW
Chip	1000 VDC
Aislamiento	2 x 4 x 5mm
Longitud de las patillas	31 mm

Tabla 2: Características de la LDR 7516

Para poder calibrar las LDR se usó la siguiente forma. Sobre uno de ellos, se le ha hecho incidir distintas intensidades de luz y a través de diversos cálculos, se ha obtenido cuanto vale la resistencia en cada momento de la LDR y su voltaje. Para saber la cantidad de luz que recibe he usado una aplicación de móvil para saber los lux que recibían. Luego a través de una gráfica, que es la Figura 9, relacionamos la Resistencia-Lumen realizada en escala logarítmica. Una vez hemos sacado algunos puntos realizamos un ajuste de mínimos cuadrados y tratamos de aproximar la recta para que la correlación lineal sea lo más alta posible:

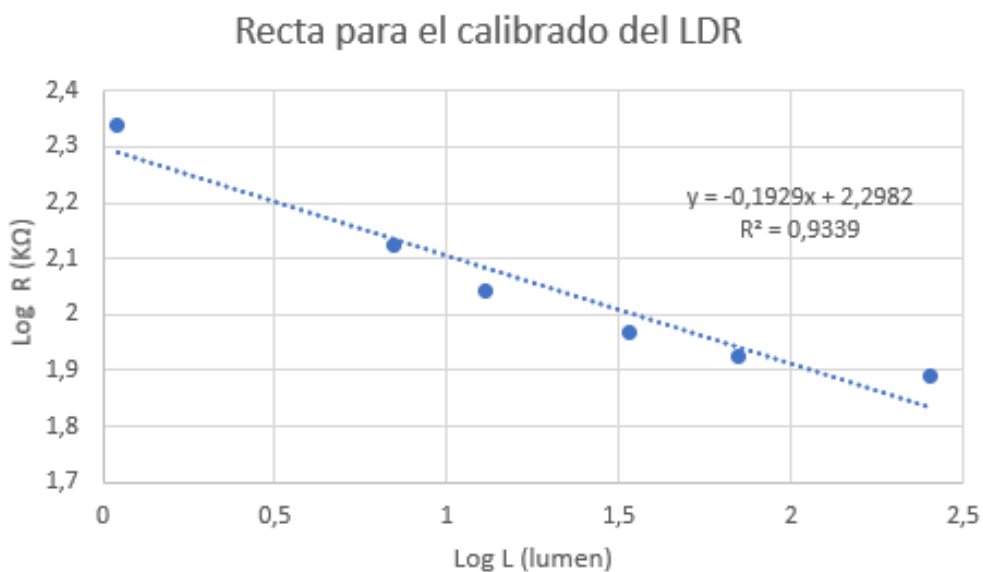


Figura 9. Recta logarítmica del calibrado de la LDR

De la ecuación obtenida a partir de la recta de la Figura 9 podemos obtener la relación de los lúmenes en función de la resistencia que presenta la LDR.

$$lm = 10^{\left(\frac{2.2982 - \log R}{0.1929}\right)}$$

2.3 Arduino Micro

El Arduino es una placa basada en un microcontrolador ATMEL, en donde podemos grabar instrucciones, las cuales se escriben con el lenguaje de programación que se puede utilizar en el entorno Arduino IDE, es de software y hardware libre. Al ser un proyecto y no un modelo concreto de placa se pueden encontrar diferentes tipos de placas según las necesidades, lo que también ha ayudado a tener una gran comunidad.

El microcontrolador tiene lo que se llama interfaz de entrada, es una conexión donde conectamos a la placa diferentes periféricos, dicha información pasa al microcontrolador que procesará lo que le llegue. Además, cuenta con una interfaz de salida que lleva la información una vez procesada a los periféricos necesarios. La programación se lleva a cabo a través del puerto serie que incorporan haciendo uso del Bootloader que traen.

Para este trabajo se ha usado Arduino Micro, el cuál podemos ver en la Figura 10 su aspecto.

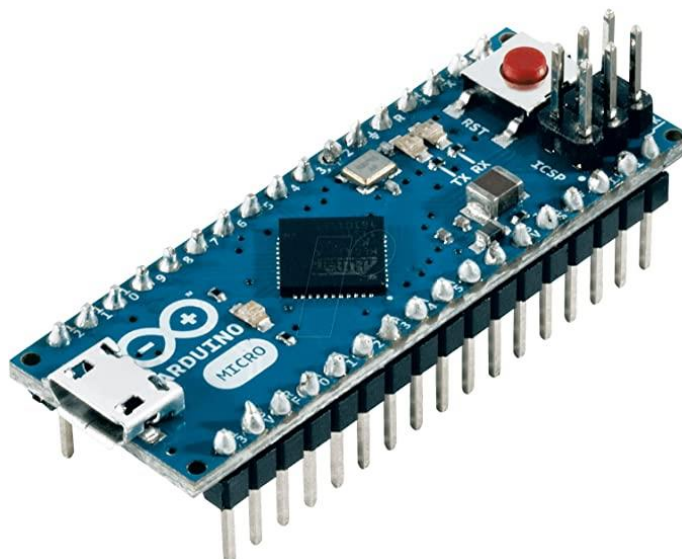


Figura 10. Arduino Micro

Voltaje de entrada	5V a 12V
Voltaje	5V
Corriente de salida max	150mA
Microcontrolador	ATMega32U4
Velocidad de reloj	16 Mhz
Entradas/Salidas digitales	12, con 5 PWM
Entradas analogicas	4
Dimensiones físicas	3.31 x 1.78 cm

Tabla 3: Características del Arduino Micro

2.4 Micro servos

Los servos son dispositivos actuadores que pueden ubicarse en una posición de su rango y mantenerse fijos en dicha posición hasta que se le ordene; pueden tener rango de 0° a 360° aunque lo normal es entre 0° a 180°.

Los servos utilizados serán: el Micro Servo SG90 y el POWER HD DSM44. Tenemos en la Figura 11 como es el SG90.

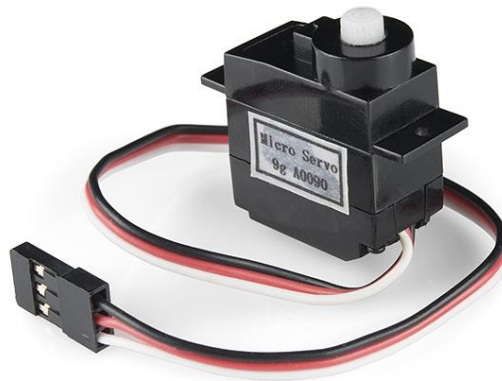


Figura 11. Micro servo SG90

Voltaje	4.8 a 6.0 V
Par	2.5 Kg-cm
Angulo de rotación	0° a 180°
Peso	14.8 g
Dimensiones	32 x 11.8 x 28.5 mm

Tabla 4: Características del Micro Servo SG90

2.5 Pan and Tilt

La estructura de Pan and Tilt sirve para poder girar objetos ya sea en el plano vertical (tilt) y en el horizontal (panning) en en este caso esta estructura tiene forma de U y gracias a los dos servos podemos girarlo y orientarlo hacia donde hayamos calculado que está el Sol. En la Figura 12 vemos el pan and tilt utilizado para este proyecto.



Figura 12. Pan and Tilt

2.6 Panel Solar

Para la alimentación del dispositivo, al principio se optó por usar pilas alcalinas AA, mediante un portapilas en serie de 4, se podía obtener 6V. Dado que va a estar en el Sol tomando muestras pensé que, para darle independencia energética, la alimentación fuese por un panel Solar.

Las placas al estar compuestas por semiconductores, al incidir sobre ellos radiación solar van desplazando los electrones sueltos de las celdas solares, más electrones vendrán del ambiente o del panel negativo que no se han ido y continuará el proceso, siendo esto una acción sin fin. En la Figura 13 podemos ver uno de reducido tamaño:



Figura 13. Panel Solar

Voltaje a Pmax	18.1 V
Corriente a Pmax	1.1A
Potencia Maxima	20W
Peso	2.1 Kg
Dimensiones	410 x 345 x 25 mm
Numero de celdas	36

Tabla 5: Características del Panel solar de 20 W

2.7 Convertidor de voltaje LM2596

El LM2596 es un regulador de tipo conmutado reductor (Step-Down), gracias a él podremos obtener a la salida una tensión constante inferior a la tensión de entrada que recibamos. Hay dos tipos de convertidores DC/DC los que son lineales y los que se les llama conmutados; en el primer grupo encontramos el LM7805, aunque son más sencillos son menos eficientes en comparación con los conmutados.

Estos últimos tienen un principio de conversión de energía basada en el almacenamiento y su posterior liberación de esa energía en el voltaje que se desea, que dese ser como máximo 1.5 V menor al de entrada para que funcione correctamente. En la Figura 14 podemos ver una pequeña placa donde hay un montaje con el LM2596.



Figura 14. Regulador LM2596

Voltaje de salida	1.23 a 37 V
Voltaje de entrada	4.5 a 40 V
Corriente máxima	3A (2.5A recomendado)
Eficiencia de conversión	92%
Dimensiones	43 x 21 x 13 mm
Potencia de salida	25W

Tabla 6: Características del convertidor regulador LM2596

2.8 Ventiladores

Una vez realizadas diversas pruebas se vio que que la temperatura que alcanzaban los componentes en un día de sol era bastante alta y podía dañar los materiales, luego teniendo en cuenta el tamaño de la caja se optó por

unos ventiladores que normalmente se utilizan para refrigerar una Raspberry Pi, pues tienen un tamaño contenido. En la Figura 15 podemos ver la forma de uno de ellos.



Figura 15. Mini ventilador

Voltaje	3.6 a 5.8 V
Corriente	0.18 A
Numero de revoluciones	13200 rpm
Volumen de aire	5m ³ /h
Dimensiones	30 x 30 x 7 mm

Tabla 7: Características del micro ventilador

2.9 Conexiones

Dado que el dispositivo iba a estar expuesto al Sol tomando muestras durante un largo tiempo y puesto que hay múltiples componentes con sus conexiones correspondientes, mantener los cables en una placa de pruebas haría muy probable que alguno de ellos se pudiera soltar. Por ello, se decidió hacer el diseño de las conexiones en una placa perforada para asegurar su estabilidad y viabilidad. Se analizó cuales eran todas las uniones que había, se optimizó su cableado para hacerlo lo más simple posible y se visualizó dichas conexiones mediante el programa Fritzing.

2.9.1 Fritzing

Es un programa libre para la automatización de diseño electrónico con el que los usuarios son capaces de pasar de diseños y prototipos a productos finales, pues permite la creación de PCBs que pueden ser compradas con los diseños hechos en el propio software. El desarrollador del proyecto fue Interaction Design Lab [5], la última versión del programa en español es la 0.9.4. Esta disponible para MacOS, Windows, Linux. El código de Fritzing está escrito en C++; posee una web para ayudar a los usuarios a compartir y discutir diseños y experiencias.

2.9.2 Diseño

Las LDRs mostradas en la Figura 16 están montadas así, pero deben ir en el soporte diseñado que se detallará en el siguiente capítulo. Para los servos, la brújula digital y el Arduino Micro se usarán pines hembra o zocalos

para encajarlos.

La placa perforada de la que se parte, era una sin pistas, luego para que el diseño fuese posible había que realizar algunas, con el el planteamiento de la Figura 16 es necesario una placa de 30x20 agujeros.

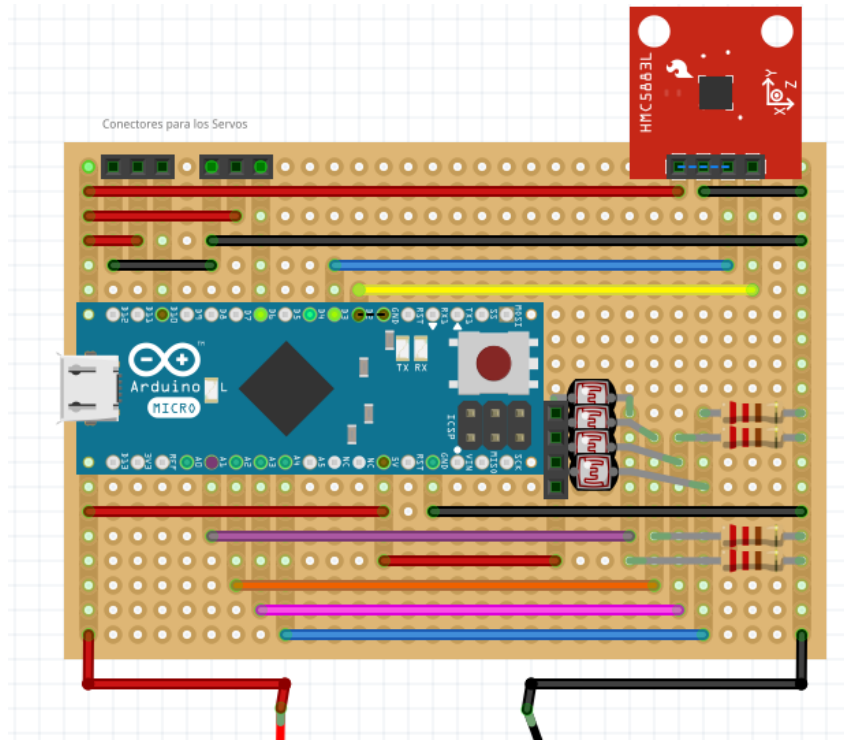


Figura 16. Conexiones

El resultado que obtiene una vez realizado es el vemos que en la Figura 17, donde ya están las conexiones realizadas y soldadas:

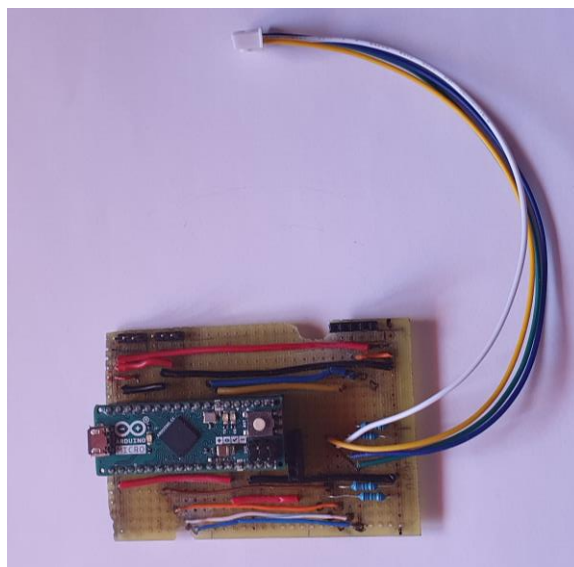


Figura 17. Placa perforada conexionada

3 DISEÑO CAD

*Una máquina puede hacer el trabajo de cincuenta personas ordinarias.
Ninguna máquina puede hacer el trabajo de un hombre extraordinario.*

Elbert Hubbard

Con el paso del tiempo y gracias a la tecnología han ido surgiendo nuevas formas de fabricar piezas, actualmente es posible hacer un diseño en un programa de software en 3D y a continuación mediante una impresora 3D hacerlo realidad y usarlo.

Las piezas ha diseñar en este trabajo han sido: una base para la sujeción del pan and tilt y un soporte para las LDRs, el cuál en los experimentos finales servirá para propiciarle sombra y mediante algoritmos de control orientaremos las fotorresistencias. Este soporte estará introducido en un tubo, al principio iba a tratarse de uno de PVC de color negro, pero, dado que me ha sido difícil encontrar un tubo cuyo diámetro exterior sea mayor de 28 mm pues, teniendo en cuenta el tamaño de las LDR 7516 y que, debe de haber unas paredes de un grosor de 3 o 4 mm a través de las cuales se encajarán las demás piezas, un diámetro interior de 24 mm era imposible que pudiera cumplir estas condiciones. Luego, en mi ciudad encontré un tubo de polietileno de alta densidad de color negro, que aquí se usa bastante para el riego y había de muchas medidas, así que me resultó fácil encontrar un tubo que se adaptará. El elegido tenía un diámetro externo de 40 mm y 35 mm interno, además el polietileno de alta densidad posee una densidad menor que el PVC (0.95 g/cm^3 vs 1.42 g/cm^3) reduciendo así el peso del conjunto.

Por último, en este trabajo, para el desarrollo CAD se han utilizado los siguientes programas: FreeCAD, SolidWorks (para retoques) y CURA.

3.1 FreeCAD

Se trata de una aplicación de diseño por ordenador en 3D, estamos ante un software libre y de código abierto basado en el software también de código abierto Open CASCADE [6] para el desarrollo de diseños 3D y que se encuentra programado en C++ y Python. FreeCAD fue desarrollado por Juergen Riegel, Werner Mayer y Yorik Van Havre la primera versión, la 0.01 fue lanzada el 29/10/2002, a día de 7/3/2021 se encuentra por la 0.19.

Presenta un entorno de desarrollo similar a los de su sector como: CATIA, SolidWorks... Tiene una arquitectura de software modular, permitiendo al usuario añadir funcionalidades sin cambiar el núcleo es por esto y por otras cualidades como la capacidad de diseñar objetos de un tamaño inimaginable por lo que se ha convertido en uno de los programas de modelado más populares del mercado. Cuenta actualmente con más de 25000 miembros activos de la comunidad solo en el foro, también es usado para simulación.

FreeCAD admite una gran gama de formatos de archivo como pueden ser STL, OBJ, SVG entre otros, además es compatible con Linux, Windows y Mac.

La creación de un modelo en 3D comienza con un boceto en 2D, el software es paramétrico, lo que quiere decir, que tanto los elementos individuales como su relación con otros se pueden ajustar, esto permitirá la creación de estructuras complejas. El boceto es visible desde varias perspectivas y modificar su tamaño, la transformación 2D a 3D se hace mediante el proceso de “relleno”, lo que se hace es colocar una capa sobre el boceto, el grosor de esta dependerá de lo que el usuario quiera, este valor se puede cambiar posteriormente sin problema, así creamos un modelo tridimensional en poco tiempo al que le podemos hacer diversas modificaciones según se quiera.

3.2 SolidWorks

Es un software de diseño CAD 3D, este posee un gran número de soluciones para distintos aspectos en el diseño y desarrollo, los productos de esta compañía nos permiten desde: crear, simular, fabricar, publicar y gestionar los datos del proceso de diseño. SolidWorks [7] fue fundada en 1993 por Hirschtick, en 1995 lanzó la primera versión del software, hoy en día la última versión disponible es la 2020 SP2.0 que data del 9/3/2020. En 1997 fue adquirida por Dassault Systemes conocida por el desarrollo del programa de diseño en 3D, CATIA.

Un paso diferenciador de este software es que mientras en las demás empresas el proceso se lleva de manera secuencial, las soluciones que hace SolidWorks permiten llevarlo de manera paralela, para ahorrar tiempo y poder tomar mejores decisiones.

Además, cuenta con un completo conjunto de herramientas, la sencillez es parte de su propuesta de valor que es decisiva para lograr el éxito de muchos clientes. Tiene también la capacidad de obtener certificaciones que permite a los diseñadores competir y sobresalir frente a otros, a parte de ser reconocidos a nivel mundial.

3.3 CURA

A parte de un programa para diseñar la pieza es preciso un software de corte o laminador, que nos permitirá convertir el modelo 3D en un archivo de código G que le dará a la impresora 3D todas las instrucciones necesarias para su elaboración. Cura fue desarrollado por David Braam en 2014 [8] y luego la adaptó la compañía Ultimaker. Es de código abierto y traducido a 15 idiomas, además del programa de este estilo más extendido en el mercado, actualmente la versión de Cura es la 4.5. Es compatible con Linux, Windows y Mac

Una de las ventajas que posee es su soporte para diversos formatos de archivos y compatibilidad con una gran variedad de impresoras 3D, admite STL, OBJ, X3D y 3MF. También podemos imprimir nuestras piezas con impresoras de otros fabricantes.

Contiene una interfaz que permite elegir entre dos modos: recomendado y personalizado. El primero es bueno para realizar la impresión con el menor número de clics y con el segundo se puede elegir entre 400 opciones de configuración, posee una vista previa para ver como se cortará el modelo.

Podemos seleccionar el filamento antes de imprimir, Ultimaker trabaja con más de 80 fabricantes de filamentos en todo el mundo, de esta manera los parámetros se modifican automáticamente según el tipo de filamento escogido. Es posible también enviar desde la nube las instrucciones de impresión directamente y con una cuenta gratis se puede acceder a todos los proyectos.

3.4 Base del pan and tilt

Para el desarrollo de esta pieza se hubo de tener en cuenta las dimensiones del servo que se utilizaba, que en el caso del pan and tilt que usamos se trata de un Micro Servo POWER HD DSM44, cuyas dimensiones se corresponden con: 20 x 9,7 x 27 mm. Al principio, la pieza se trataba de una caja hueca, en donde en la parte superior se encuentra un hueco donde encajar el servo, además de otros dos para posibles tornillos o cables.

Para poder entrar al interior de la base pensé en hacer que una de las paredes laterales estuviese conectada mediante una bisagra para poder tener libertad de movimiento y acceder, pero Juan de Dios Yañez, al cual le pedí consejo e información durante el proceso del diseño en 3D, me recomendó que la tapa para acceder al interior fuese la pared inferior de la caja, más que uno de los laterales, así sería más cómodo. También me aconsejó que si usaba bisagras atornillables hiciera el guiado del tornillo porque se puede prevenir así que el plástico se rompa, una primera versión de este diseño vemos en la Figura 18 la parte superior de la caja y como se vería desde una vista inferior.

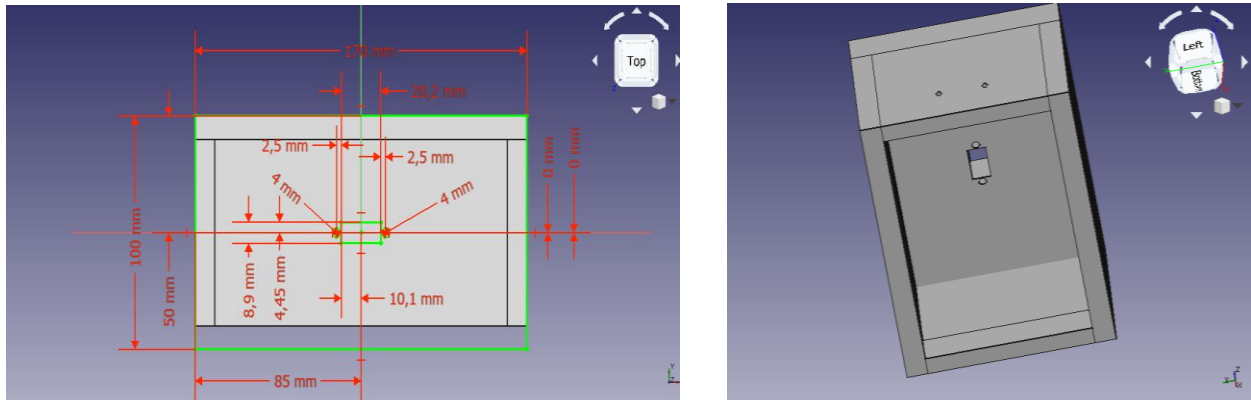


Figura 18. Distintas vistas de la pieza preliminar

La parte inferior de la pieza estaría formada únicamente por una pared con un determinado grosor, y dos agujeros para los tornillos que la unirían con la bisagra.

Este era el diseño del cuál iba a partir para imprimir, pero la persona que me ayudó a la realización de la impresión 3D me comentó las limitaciones y problemas que tenía este diseño y cuál pensaba que sería un diseño mejorado:

- Las dimensiones de la caja limitan el peso que se le puede añadir como apoyo, si este peso no fuese suficiente sería necesario rehacer la caja, hacerla más grande, gastar más material y más tiempo.
- Teniendo en cuenta las dimensiones del servomotor, la caja supera esa longitud en 8,5 veces de largo y 11,1 veces en ancho que era poco óptimo.
- Las paredes tenían un grosor desorbitado, en el software de corte CURA, se vió que el tiempo que se emplearía para fabricar esta pieza sería de 19h y 30 min, al cual habría que sumar el de la parte inferior.

Para mejorar el diseño, el me ayudó a enfocar la pieza desde otro punto de vista. Esta nueva pieza, solo estaría formada por una única parte, más estrecha, más delgada y ligera, una imagen de esta nueva pieza está en la Figura 19:

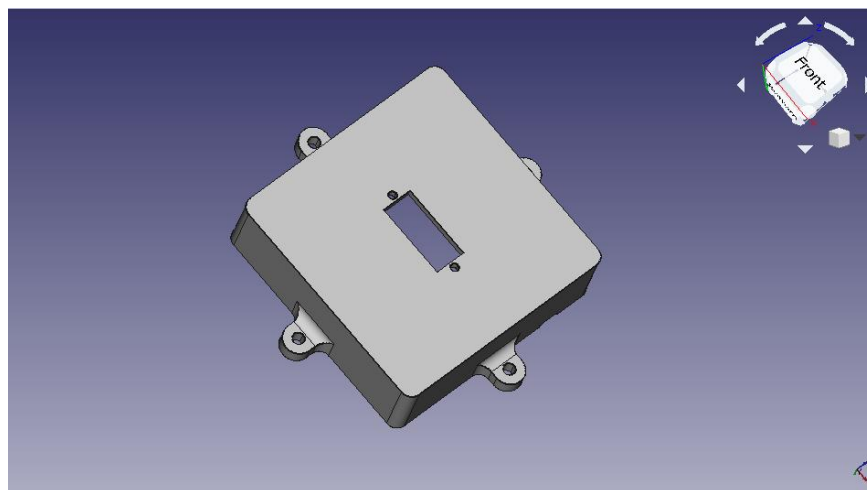


Figura 19. Diseño segundo de la base preliminar

Con este diseño no tenemos las limitaciones comentadas, pues, la caja es de 6.4 x 6.4 x 1.9 cm, estamos hablando de una cuarta parte de superficie superior, el grosor de las paredes baja a 3mm con lo que el tiempo de fabricación baja a 4h y 30 min. Con este diseño, no tenemos el problema de no poder tener el suficiente apoyo, pues esta pieza irá atornillada a un tablón de madera el cuál siempre se podrá ampliar sin tener que rehacer la pieza.

Debido al diseño de las conexiones, se decidió que estas en lugar de estar estos en una caja auxiliar, en la cual saliesen los cables de entrada y salida necesarios, se adaptó el diseño final de la caja para hacer que pudiera albergar la placa, el Arduino y el resto de los componentes.

Para ello se partió del diseño comentado en la Figura 19 y se separó este en dos; una primera con forma de L en donde estaría la placa alojada en la parte inferior y un agujero para acoger el servo del pan and tilt, una imagen de esto lo podemos ver en la Figura 20.

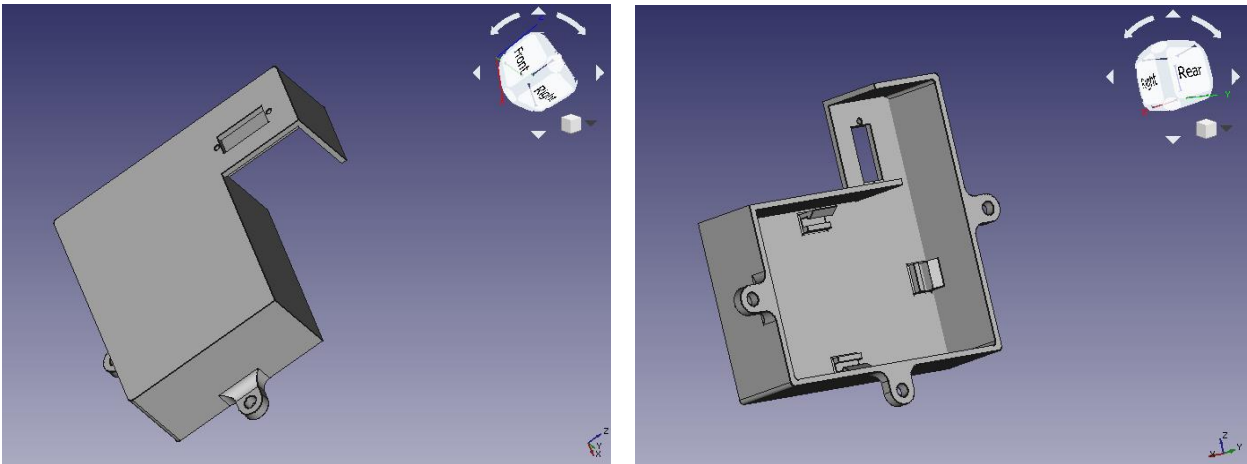


Figura 20. Diseño final de la base

La caja tiene unas dimensiones externas de 101 x 93 x 32.5 mm las paredes son de 2 mm mientras que la cubierta es de 2.5 mm para darle mayor apoyo. La alimentación externa va alojada en el espacio que falta para completar el rectángulo, se trata de un hueco para alojar 4 pilas de tamaño AA en formato de portapilas. En la Figura 21, vemos como queda una vez impreso y pintado de color blanco, para que los componentes no sufran por la temperatura, ha sido necesario hacer un agujero para sacar los cables:



Figura 21. Impresión de la base

3.5 Soporte LDR.

La utilidad de esta pieza es alojar los LDR para que durante las primeras pruebas a bucle abierto ir comprobando los valores de la radiación obtenidos. Luego debe de tener huecos para poder alojar los LDRs y para pruebas posteriores debe de tener alguna pared que les haga sombra para poder orientarlos con algoritmos posteriores.

Partiendo de esta idea y de que la pared que haga sombra debe de tener una altura que de primeras no se sabe cual será la correcta, luego debe de tener diferentes alturas y poder intercambiarse, en la Figura 22 podemos ver los dos tipos de piezas que eran al principio:

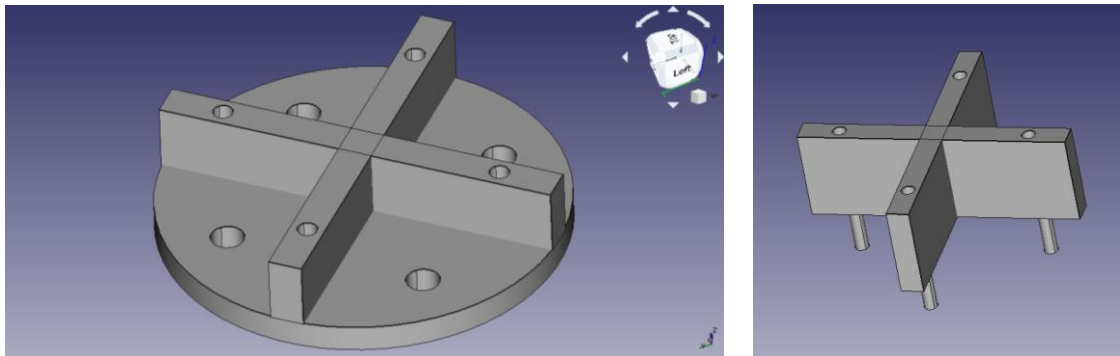


Figura 22. Soporte LDRs y pared extraíble preliminar

Así, si se hace con diferentes alturas se podrían encajar una encima de otra y combinarlas de manera que se pudiera encontrar un valor correcto. Sin embargo, al comentarle esto a Juan de Dios, me explicó que no era viable pues esas guías delgadas y alargadas en una altura superior a 5 mm suelen ser frágiles y luego su vida útil se reduciría considerablemente.

Luego, me recomendó enfocarlo de otra manera dándome la idea de hacer que las superficies se quedasen encajadas en lugar de clavar una sobre otra y que la unión sea una varilla metálica algo que se puede apreciar en la Figura 23 donde se puede observar como se atornillan y se vería:

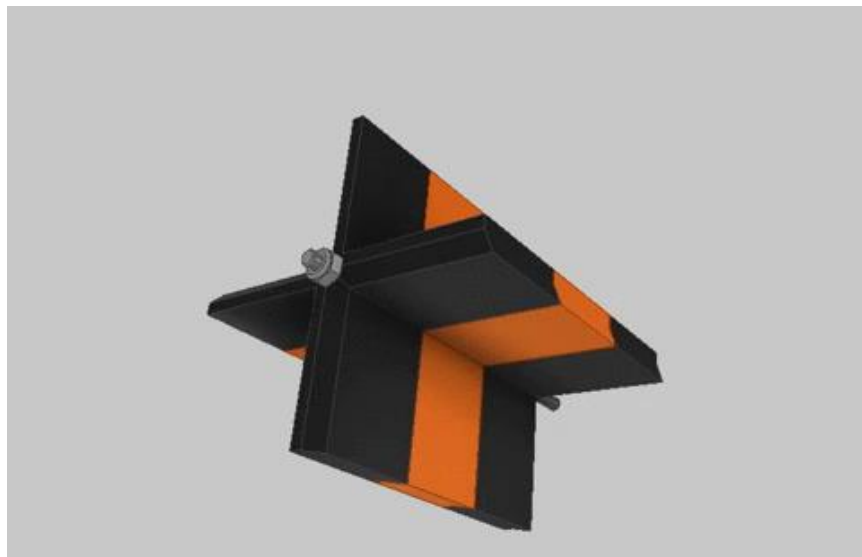


Figura 23. Encaje de las paredes

Para poder verlo más cerca, Juan de Dios me pasó la siguiente imagen que podemos ver en la Figura 24, donde vemos los perfiles de las paredes y como sería el diseño para que estos encajen, además de su interior formado por una varilla metálica que los atravesaría y con una tuerca autobloqueante para que se quedasen fijas y no deslizaran.

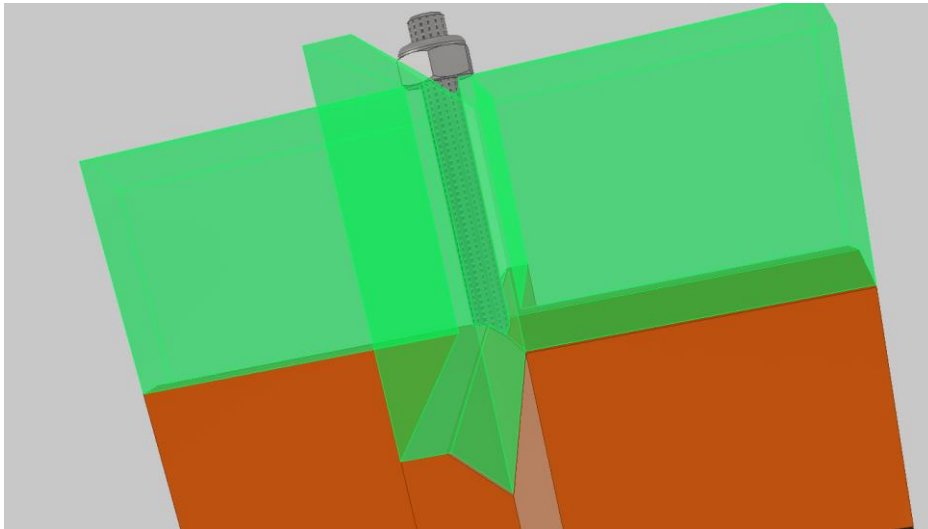


Figura 24. Perfil e interior de la pieza

Ambos videos me los proporcionó Juan de Dios. Una vez ya con la idea, comencé a diseñar las piezas, me ajusté a lo que él me comentó y el resultado es lo que se puede observar en la Figura 25, se ha añadido un encaje con respecto al diseño inicial para que pueda ir el tubo de polietileno y no entre irradiación difusa o reflejada y tomar las medidas lo más precisas posibles.

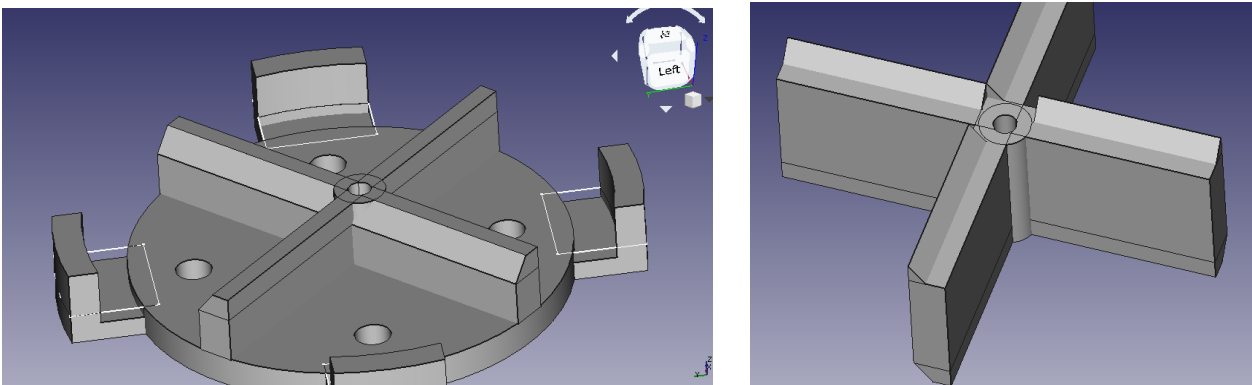


Figura 25. Diseño segundo del soporte y pared preliminar

La imagen de la derecha presenta una altura de 2 cm, pero la idea es realizar varias con distintos diámetros y así poder ajustar la altura. Al ir a la persona que me imprimió las piezas me sugirió un par de modificaciones para mejorar la pieza y que se ajuste mejor.

Estas diferencias se pueden apreciar en la Figura 26, el soporte en lugar de tener meros agujeros para pasar las patillas de las LDRs tiene la figura de estas para poder encajarlas y reducir su altura, además el perfil de las paredes ha cambiado, al hacerlo en forma de semicircunferencia la fuerza podrá distribuirse mejor y se quedarán bien encajadas las piezas.

Las dimensiones del soporte son de 35 mm de diámetro, las paredes poseen un grosor de 3.6 mm, una altura de 2 mm, quedando así en su parte más alta una altura de 3.8 mm y la parte más baja de 0.2 mm, estas paredes deben ser pequeñas para no proporcionar sombra durante los experimentos e interferir en las medidas.

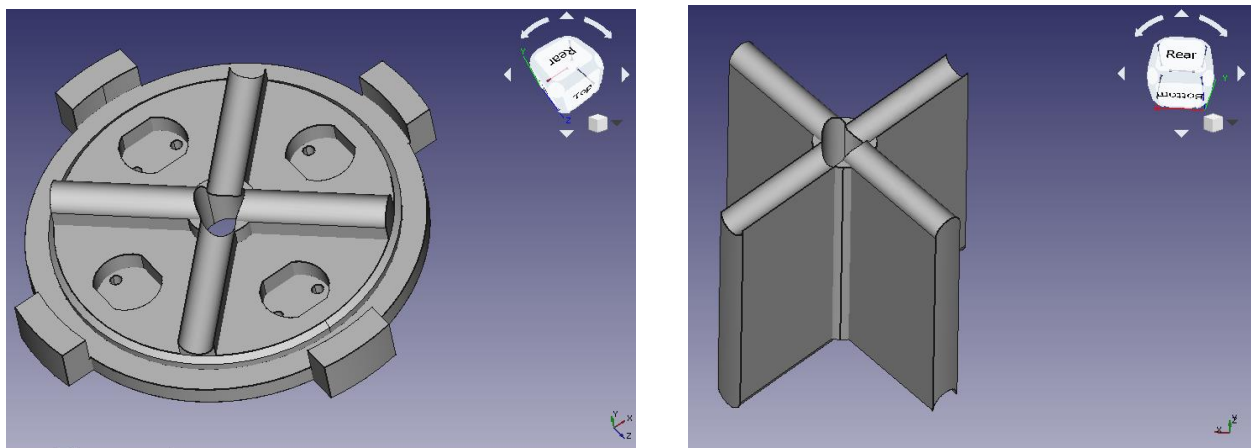


Figura 26. Diseño final

Una vez ya diseñadas y verificadas las piezas, se imprimieron y en la Figura 27 se puede observar como han quedado.

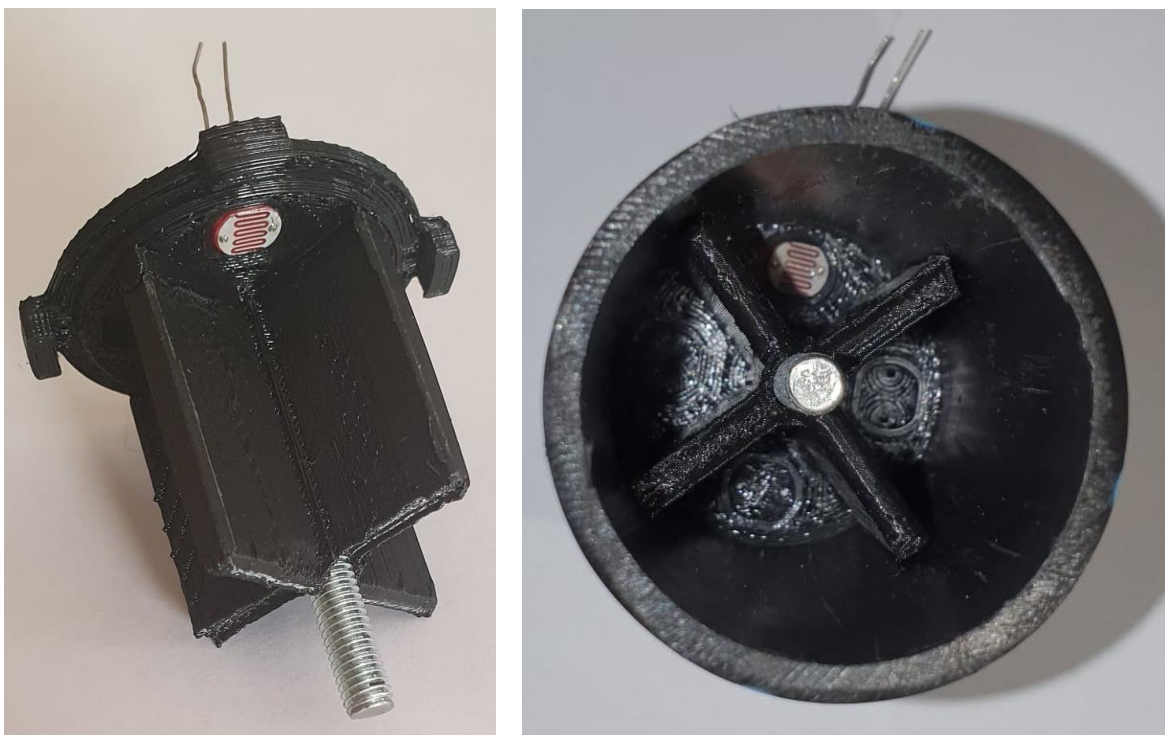


Figura 27. Impresión Final

4 ECUACIONES SOLARES

4.1 Introducción

Aquí se tratará de explicar como se puede calcular con ecuaciones la altitud y el acimut del Sol [9]. Con la Figura 28 nos podemos orientar, para empezar, calcularemos las coordenadas de un vector, S, que apunta al Sol y parte del centro de la Tierra, en el sistema de referencia N, E, Z que se sitúan sobre la superficie de la Tierra. Estas coordenadas estarán en función de los ángulos ω (ángulo horario), δ (declinación), λ (latitud). Ahora haremos que estas coordenadas dependan y estén en función de la altitud y el acimut.

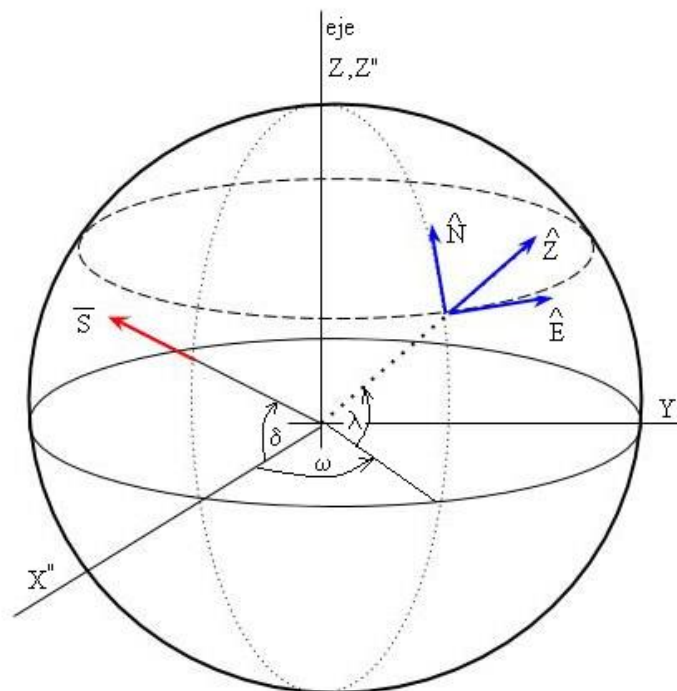


Figura 28. Vector S en el sistema de referencia con centro en la Tierra

En este sistema de referencia en donde i'' , j'' , k'' son los vectores unitarios, el vector S que va en la dirección del Sol, se encuentra en el plano X'' , Z'' :

$$S = \cos(\delta) * i'' + \sin(\delta) * k''$$

El ángulo diario τ , expresado en radianes, viene dado por la siguiente expresión:

$$\tau = \frac{2\pi}{365} (d_n - 1)$$

En donde d_n se trata del día juliano del año, este irá desde 1 hasta 365 para el 31 de diciembre. La declinación se puede decir que se trata del ángulo que forman el plano ecuatorial de la tierra y la línea que une los centros del sol y de la tierra, pues esta línea cambia cada día, incluso cada instante. La declinación es 0 en los equinoccios y de $\pm 23.5^\circ$ durante los solsticios:

La ecuación que la calcula, en grados, viene dada por la siguiente fórmula de Spencer y en la Figura 29 podemos ver cual es la variación de la declinación en función del día en el que nos encontremos:

$$\delta = (0.006918 - 0.399912 \cos\tau + 0.070257 \sin\tau - 0.0067 \cos 2\tau + 0.0009907 \sin 2\tau - 0.002697 \cos 3\tau + 0.00148 \sin 3\tau)$$

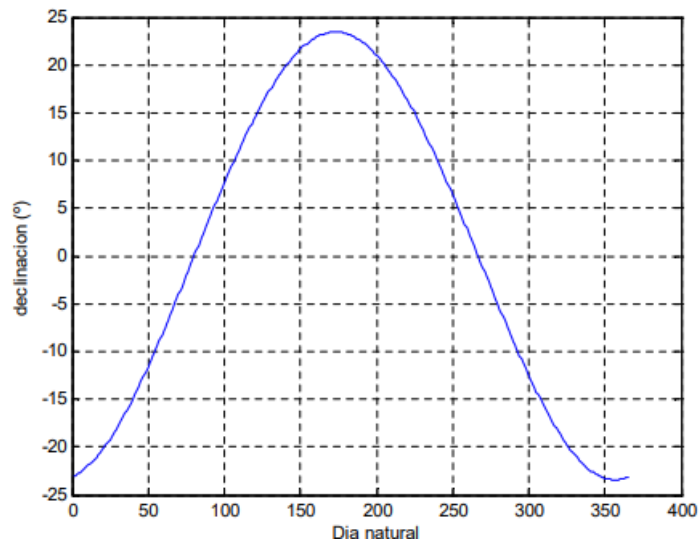


Figura 29. Variación de la declinación

El punto sobre la superficie terrestre está definido por los ángulos λ (latitud) y ω (ángulo horario). Para el ángulo ω es preciso conocer que es la ecuación del tiempo E_t [10].

El tiempo solar verdadero, se basa en la rotación de la Tierra sobre su eje polar y el movimiento de traslación alrededor del Sol. Un día solar es el intervalo de tiempo en el que el Sol completa un ciclo alrededor de un observador estacionario que esté en la Tierra; la velocidad de la Tierra no es uniforme a lo largo del año, es por ello por lo que el día solar no es uniforme. Para evitar esto, se toma una esfera terrestre ficticia que posee un movimiento de rotación uniforme alrededor del Sol, así surge el término de tiempo solar medio.

A la diferencia entre el tiempo solar verdadero y el tiempo solar medio es la ecuación del tiempo, que varía de un lugar a otro, Spencer propone la siguiente fórmula para poder calcular la ecuación del tiempo:

$$E_t = (0,000075 + 0,001868\cos\tau + 0,032077\sin\tau - 0,014615\cos2\tau - 0,04089\sin2\tau)(229,18)$$

El término de la izquierda es para la ecuación del tiempo en radianes, mientras que 229.18 es para convertirlo a minutos, el error máximo de esta ecuación es de 0.0025 radianes o 35 seg. En la Figura 30 se puede apreciar la variación anual de la ecuación del tiempo.

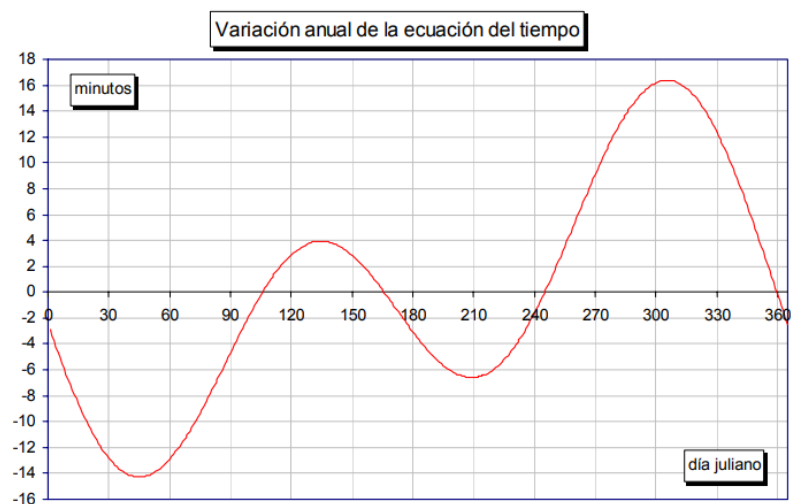


Figura 30. Alteriación de la ecuación del tiempo

La hora local aparente puede expresarse de la siguiente manera

$$T_s = \text{tiempo local estandar} + \text{correccion de longitud} + E_t$$

Siendo la corrección de la longitud; 4·longitud y el tiempo local estándar, la hora del día. Una vez ya tenemos descritas las ecuaciones que nos relacionan todo ya podemos ver como se calcula el ángulo ω mediante la siguiente fórmula:

$$\omega = ((Ts - 12) * 15) * (\pi/180)$$

Ya podemos seguir tratando de expresar S en el sistema de referencia de N, E, Z. Expresamos E, N, Z, en términos de los vectores unitarios i'' , j'' , k'' , en la Figura 31 está la descomposición de E en el sistema de referencia

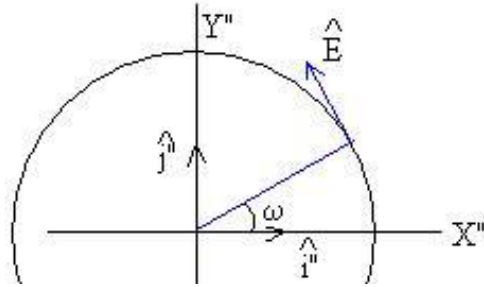


Figura 31. Vector E en el sistema ij

Es fácil apreciar en la imagen que $E = -\sin\omega * i'' + \cos\omega * j''$. En la Figura 32

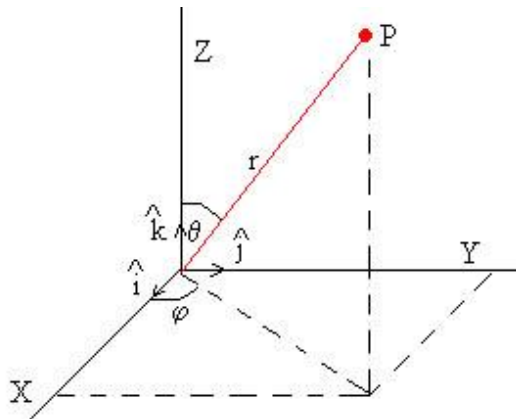


Figura 32. Punto representado en el sistema XYZ

Gracias a la Figura 32 podemos expresar el vector r en coordenadas esféricas y como Z está en la dirección radial se puede expresar como:

$$Z = \cos\lambda \cdot \cos\omega \cdot i'' + \cos\lambda \cdot \sin\omega \cdot j'' + \sin\lambda \cdot k''$$

Luego ya con Z y E podemos calcular N como el producto vectorial de E x Z

$$N = \begin{vmatrix} i'' & j'' & k'' \\ \cos\omega\cos\lambda & \cos\lambda\sin\omega & \sin\lambda \\ -\sin\omega & \cos\omega & 0 \end{vmatrix}$$

$$N = -\cos\omega \cdot \sin\lambda \cdot i'' - \sin\omega \cdot \cos\lambda \cdot j'' + \cos\lambda \cdot k''$$

Una vez que ya tenemos los vectores podemos sacar los componentes del vector S gracias a los ejes E, N, Z mediante el producto escalar de estos por el vector S

$$S_N = S \cdot N = -\cos\omega \cdot \sin\lambda \cdot \cos\delta + \cos\lambda \cdot \sin\delta$$

$$S_E = S \cdot E = -\sin\omega \cdot \cos\delta$$

$$S_Z = S \cdot Z = \cos\omega \cdot \cos\lambda \cdot \cos\delta + \sin\lambda \cdot \sin\delta$$

Luego una vez todo junto, obtenemos el siguiente vector:

$$S = -\sin\omega \cdot \cos\delta \cdot E + (\cos\lambda \cdot \sin\delta - \cos\omega \cdot \sin\lambda \cdot \cos\delta) \cdot N + (\cos\omega \cdot \cos\lambda \cdot \cos\delta + \sin\lambda \cdot \sin\delta) \cdot Z$$

4.2 Acimut y altitud

Antes de calcular cualquiera de estos ángulos a partir de lo que hemos sacado, trataremos de explicar y entender que es cada uno.

4.2.1 Acimut

Es el ángulo que forma el Norte y el cuerpo, medido en el sentido de rotación de las agujas de un reloj alrededor del horizonte de la persona. Este ángulo nos permite determinar la dirección de un cuerpo celeste, es decir, si este está en el Norte su acimut será de 0°, al Este 90°, al Sur 180° y al Oeste 270° algo que se puede ver en la Figura 33.

En los equinoccios el Sol sale directamente desde el este y se pone directamente al oeste, pero, en general, el ángulo de acimut varía con la latitud y la época del año.

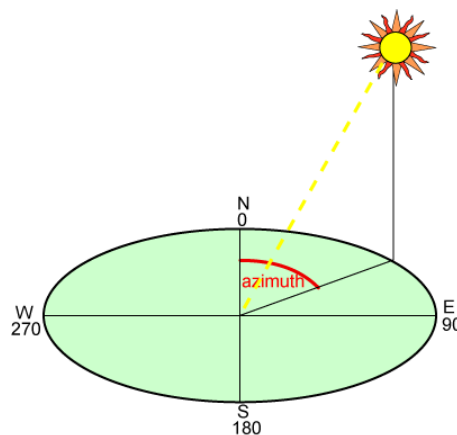


Figura 33. Ángulo acimut

4.2.2 Altitud o elevación

Es el ángulo que se forma entre el cuerpo celeste que queremos y el horizonte de la persona que lo está observando o el plano local del observador. La elevación del Sol la podemos tomar como el ángulo que forma la dirección de su centro con el horizonte local nuestro tal y como se representa en la Figura 34.

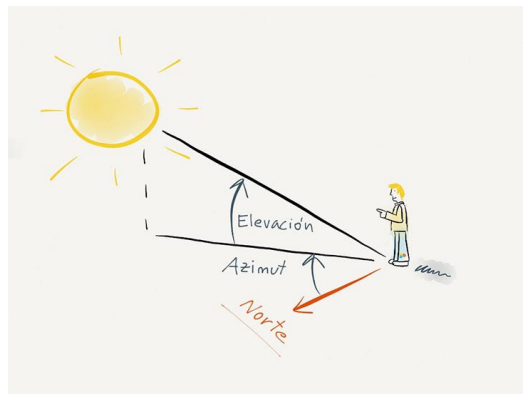


Figura 34. Elevación

4.2.3 Cálculo

Una vez vistos que son ya podemos calcular estos ángulos, en la Figura 35 podemos ver que S se puede expresar en función de estos ángulos.

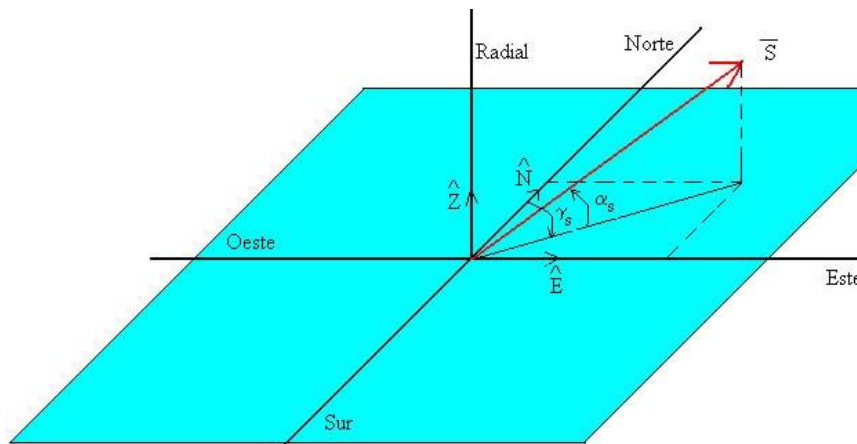


Figura 35. Vector S con sistema de referencia local

Entonces, el vector S se puede describir:

$$S = \cos \alpha_s \cdot \sin \gamma_s \cdot E + \cos \alpha_s \cdot \cos \gamma_s \cdot N + \sin \alpha_s \cdot Z$$

Ahora, si igualamos lo que tenemos con lo que ya obtuvimos de antes para el vector S expresado en los ejes N, E, Z podemos obtener:

$$\cos \alpha_s \cdot \sin \gamma_s = -\sin \omega \cdot \cos \delta$$

$$\cos \alpha_s \cdot \cos \gamma_s = \cos \lambda \cdot \sin \delta - \sin \lambda \cdot \cos \omega \cdot \cos \delta$$

$$\sin \alpha_s = \cos \lambda \cdot \cos \omega \cdot \cos \delta + \sin \lambda \cdot \sin \delta$$

Gracias a la última ecuación podemos despejar la altitud (α_s):

$$\sin \alpha_s = \cos \lambda \cdot \cos \omega \cdot \cos \delta + \sin \lambda \cdot \sin \delta$$

Ahora de la penúltima ecuación podremos sacar el acimut (γ_s) como:

$$\cos \gamma_s = \frac{\cos \lambda \cdot \sin \delta - \cos \omega \cdot \sin \lambda \cdot \cos \delta}{\cos \alpha_s}$$

Para calcular el acimut necesitamos calcular el arco coseno que solo nos devolverá valores entre 0° - 180° , pero, como se ha visto, el acimut varía entre 0° - 360° luego para estos casos se utiliza la siguiente regla:

- Si el ángulo horario $\omega < 0$ entonces el acimut se mantiene igual
- Si el ángulo horario $\omega > 0$ entonces el acimut será $\gamma_s = 360 - \gamma_s$

Si el acimut está comprendido entre 0° - 180° el Sol estará al Este y será con las horas entre medianoche y mediodía, mientras que si pertenece al intervalo de 180° - 360° el Sol estará al Oeste del observador y estaremos en las horas que hay entre mediodía y medianoche.

5 EXPERIMENTOS

En el presente capítulo se explicarán las diferentes pruebas realizadas y los resultados que se han ido obteniendo. Las pruebas se dividirán en dos grandes bloques: el primero estará formado por los distintos experimentos con los tubos o sin ellos en bucle abierto; mientras que el segundo está constituido por varios algoritmos de búsqueda para mejorar los datos obtenidos al principio.

Se almacenarán los valores de los cuatro sensores en forma de porcentaje con respecto a un valor máximo que se tomó con el flash del móvil. Así será posible usar un byte por cada medida en cada minuto y por cada sensor, ahorrando espacio en la memoria del Arduino, dichos valores se almacenan en una tabla cuya longitud es de un máximo de 330 min (dependiendo del programa a ejecutar). A continuación, los valores se pasan a un txt y mediante un script de Matlab se saca una gráfica con dichos valores en función de la hora y día en el que se encuentre.

En la Figura 36 se puede ver como queda el montaje del dispositivo una vez colocado, atornillado y montado en la base del pan and tilt:

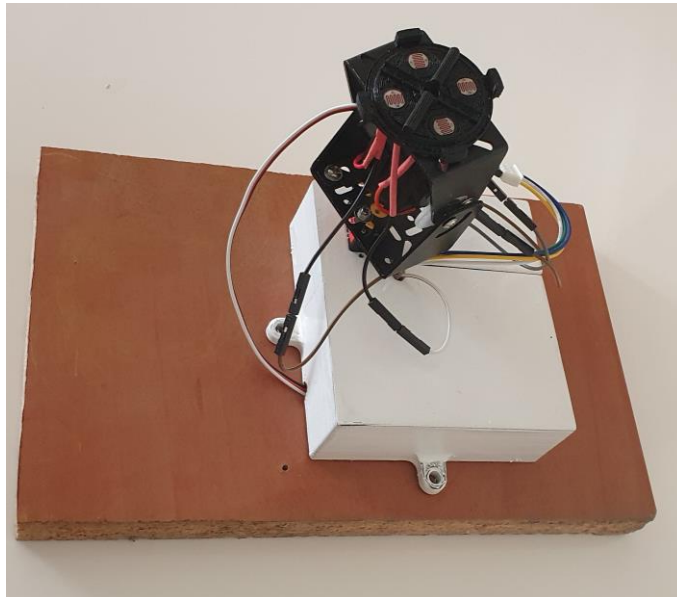


Figura 36. Montaje completo del seguidor solar

5.1 En bucle abierto

Al principio, el planteamiento que se propuso era realizar cinco tipos de pruebas aprovechando los cuatro tubos de los que disponía con distintas longitudes y un primer experimento sin tubo para comprobar la diferencia. Más adelante se detallará porque esto no acabó siendo viable y se cambiaron las pruebas.

5.1.1 Medición sin tubo

El objetivo aquí era tomar los resultados de esta prueba para compararlos con los que se realizarán con los tubos y poder ver como influye la irradiancia difusa y la reflejada. En la gráfica mostrada en la Figura 37 se observa el comportamiento de los sensores a lo largo de un día:

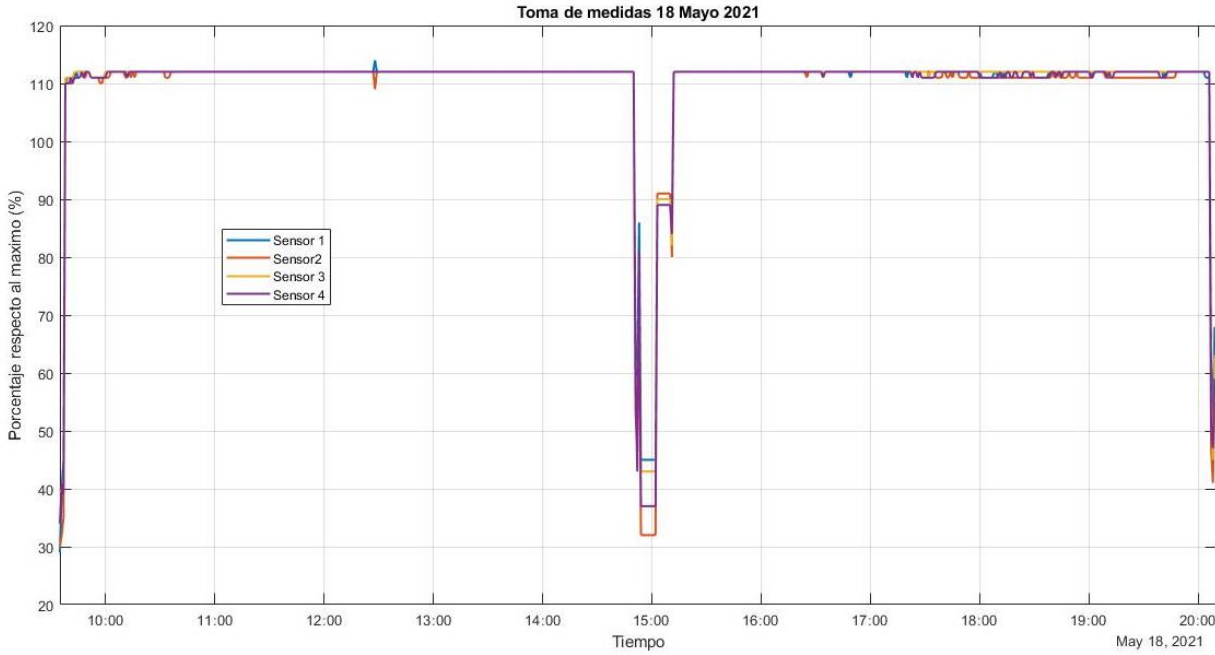


Figura 37. Medidas del 18 de mayo

Podemos ver en la Figura 37, que primera hora de la mañana hasta última hora, tienen un valor prácticamente uniforme y sin variaciones cercano a 110. Debido a la escasa memoria del Arduino las pruebas se dividen en dos y es en el período de las 14:40h-15:05h, de ahí la drástica caída de valores ya que en ese tiempo debía de recoger el aparato de la azotea, bajar al piso, volcar los datos con el puerto serie en un txt, programarlo con la nueva hora y subirlo de nuevo a la azotea.

Este no era el resultado que se esperaba al realizar la prueba, pues, al principio se pensaría que la gráfica se asemejaría más a la que se puede observar en la Figura 38, una curva en donde en las horas de menor irradiación solar las LDRs captasen unos valores más bajos y conforme pasase el tiempo, este valor se incrementase hasta un pico, para por la tarde que estos indicadores comenzasen a bajar.

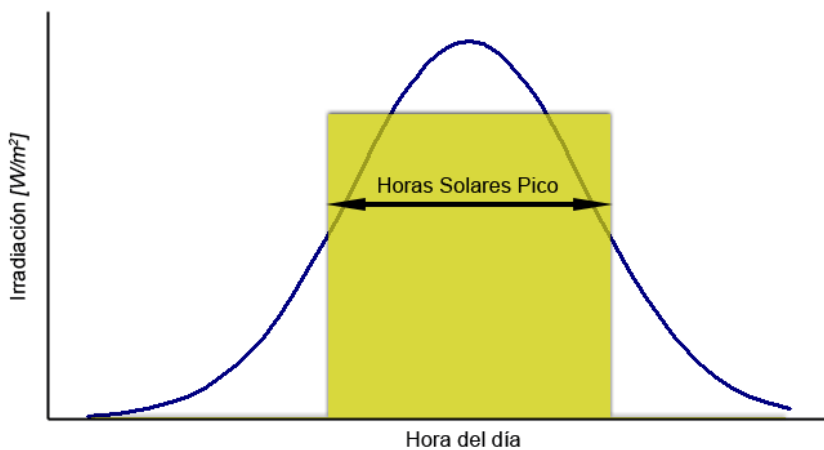


Figura 38. Curva de Irradiación Solar

Después de realizar un par de pruebas más, y consultando con los tutores, se concluyó que desde primeras horas de la mañana las fotorresistencias saturaban. Así se pudo comprobar en un breve experimento en donde a las 9:00h se mostraba el valor que arrojaba el AnalogRead() de cada sensor, este valor está comprendido entre 0 y 1023 al tener el Arduino Micro ADC de 10 bits. Y su valor se encontraba en 1017-1018 casi en su

máximo desde primera hora.

Además de esta gráfica, en la Figura 39 se puede observar la suma de todas las fotorresistencias a lo largo del tiempo:

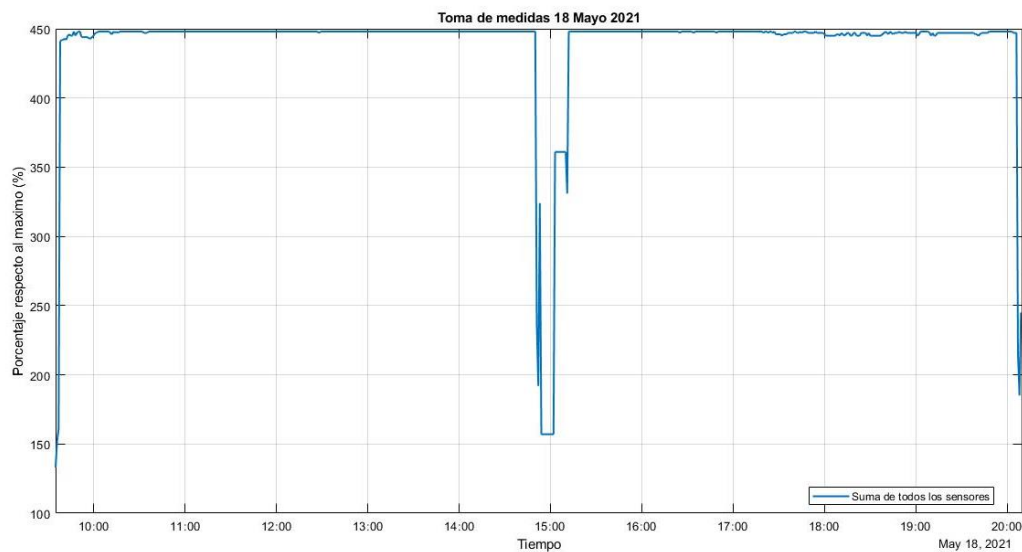


Figura 38. Suma de mediciones 18 de mayo

Al tener cada una un valor cercano a 111 o 112 la suma da casi 450 salvo en un periodo. También de esta gráfica se pueden sacar datos que nos pueden ser útiles para comparar, como puede ser la media del vector compuesto por la suma de los 4 sensores y la desviación estándar (en las pruebas cuya duración sea todo el día, se aportarán los datos totales y los correspondientes a la mañana para disponer de todos los datos necesarios):

Media: 438.6740

Desviación estándar: 45.9952

Media parcial: 441.6125

Desviación estándar parcial: 43.0130

La media a lo largo del día es algo más baja que la de la mañana, debido principalmente a la caída experimentada a final de esta; la desviación estándar se comporta a la inversa, disminuye levemente en el día total respecto a la mañana.

5.1.2 Medición con tubo de 10 cm

En este caso se esperaba que, al no tener irradiación difusa y reflejada, las LDRs no saturasen, pero no se pudo llevar a cabo este experimento ni los sucesivos con los distintos tubos, pues el peso de estos era tan grande que el pan and tilt se desequilibraba y perdía bastante altitud. Además, de que al estar unido el pan and til solo por el engranaje del servo, éste se apoyaba sobre un eje muy pequeño que lo hacía realmente inestable.

Entonces, se cambió de material para realizar los tubos de las mediciones y se empleó un plástico flexible de color, en la Figura 40 se puede ver cual es su resultado:



Figura 40. Tubos de plástico

Para tener una idea, comprobé cual era el peso del tubo de 10 cm de plástico y su correspondiente en PVC. Eran bastante diferentes: 4g vs 29 g, respectivamente, supone una reducción de peso del 74%. También, se realizaron algunas modificaciones y ajustes para que el servo no tuviera tanto movimiento. De estos tubos hay dos longitudes, de 5 y de 10 cm.

5.1.3 Filtro óptico

Por otro lado, para solucionar el problema de la saturación de las LDRs se decidió hacer uso de un filtro óptico que pudiese bloquear parte de la irradiancia que llegaba a estas. El elegido fue un filtro polarizado, se usarían dos trozos de este sobre los tubos y se girarían en función de la irradiancia que se quiera bloquear para obtener un resultado u otro, un ejemplo de este tipo de filtro se puede ver en la Figura 41:

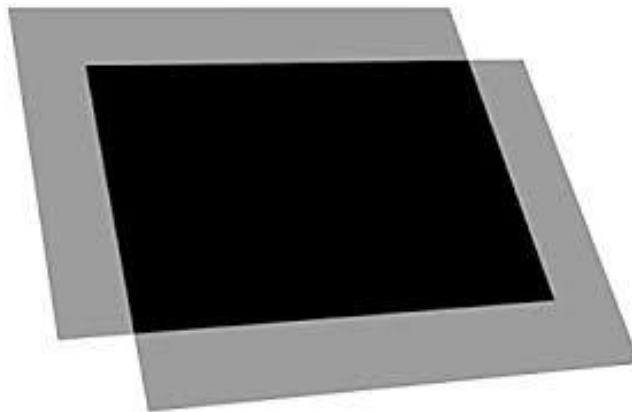


Figura 41. Filtro óptico

Una vez ya se tienen los tubos y el filtro se puede continuar con las pruebas.

5.1.4 Medición con tubo de 5 cm y filtro óptico

Aquí se probará el tubo de menor longitud y las capas del filtro estarán giradas unos 30° aproximadamente, con este giro haciendo pruebas con el flash del móvil se obtenían unos valores máximos entorno a 90, para que así

no saturen, la Figura 42 que muestra la toma de datos durante la mañana del 2 de junio nos permite observar lo siguiente:

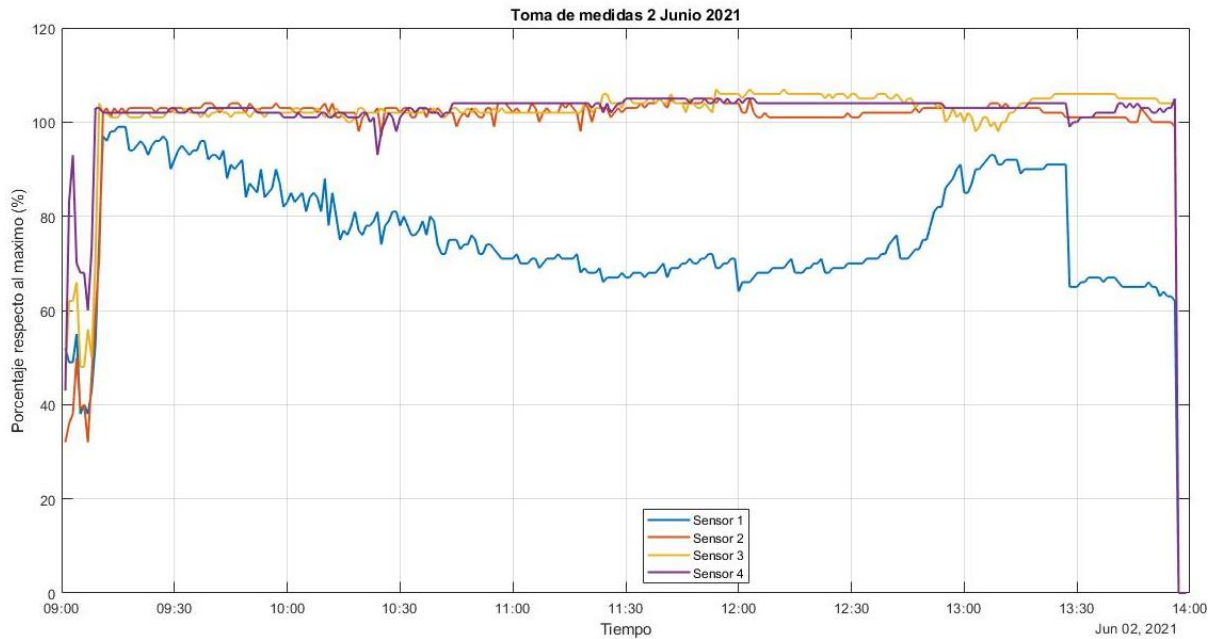


Figura 42. Medidas del 2 de junio

- Se puede ver que ya las LDRs no permanecen en el mismo valor que en el caso inicial.
- Al introducir el tubo y el filtro y no estar orientado de forma exacta al Sol, puesto que los servos o los cálculos tienen un ligero error, los 4 sensores no obtienen las mismas medidas como lo hacían al principio. Al sensor 1 le llega menos irradiancia que a los demás.
- El comportamiento sigue sin ser el deseado puesto que, aunque no saturan al valor de 110, si que saturan en uno más bajo, 105. Luego habrá que modificar las posiciones relativas de las capas del filtro para que bloquee más irradiancia y bajar más estos valores.

Si observamos la Figura 43 podremos ver el comportamiento de la suma de los 4 sensores en la mañana:

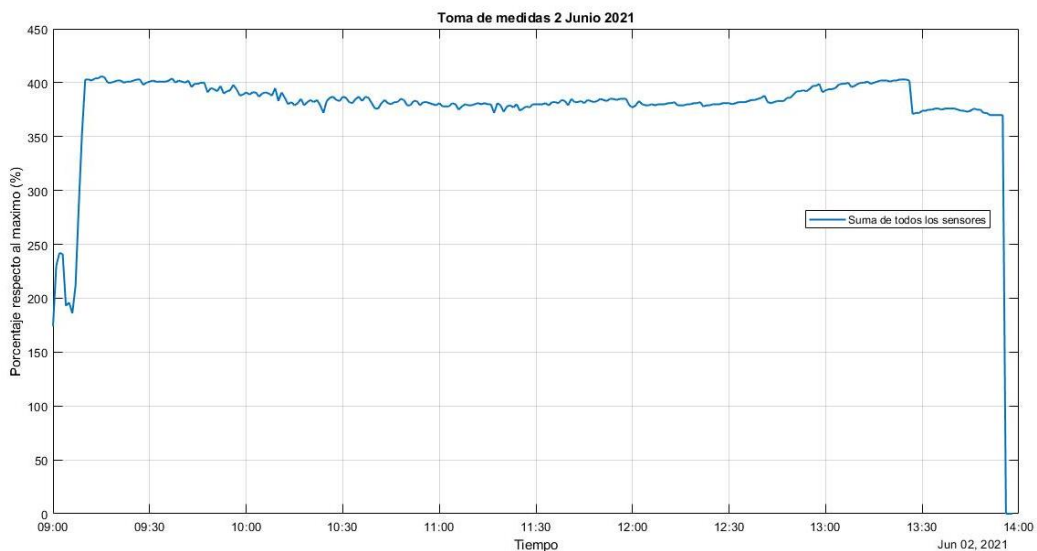


Figura 43. Suma de medidas 2 de junio

Los valores extraídos matemáticos extraídos de esta gráfica son los siguientes:

Media: 377.0736

Desviación estándar: 48.9592

Se puede ver que la media disminuye por el descuadre del sensor 1 y que debido a las variaciones la desviación aumenta respecto al experimento anterior.

5.1.5 Medición con tubo de 10 cm y filtro óptico

Se realizó una nueva toma de medidas, esta vez empleando el tubo fabricado de 10 cm para ver que cambios se podían apreciar respecto al caso anterior y en la Figura 44 podemos ver el resultado:

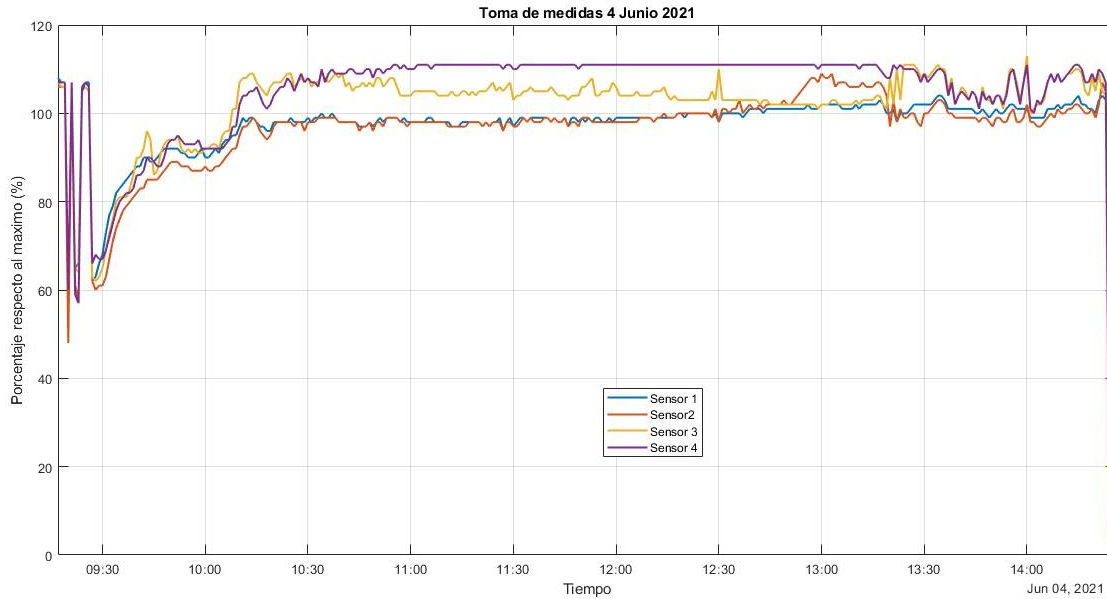


Figura 44. Medidas del 4 de junio

Antes de analizarlo, he de comentar que el viernes 4 de junio hubo varias nubes, como se puede apreciar en el descenso de las lecturas de las LDRs entre las 9:30h y las 10:15h, y en el tramo de las 13:15h a 14:10h, siendo este último menos significativo.

- A parte de lo ya comentado, se presenta una situación bastante similar a la anterior, en la que la saturación baja en algunos sensores, pero se sigue manteniendo estable entorno al valor de 107.

La Figura 45 nos mostrará la suma de las LDRs en esta ocasión:

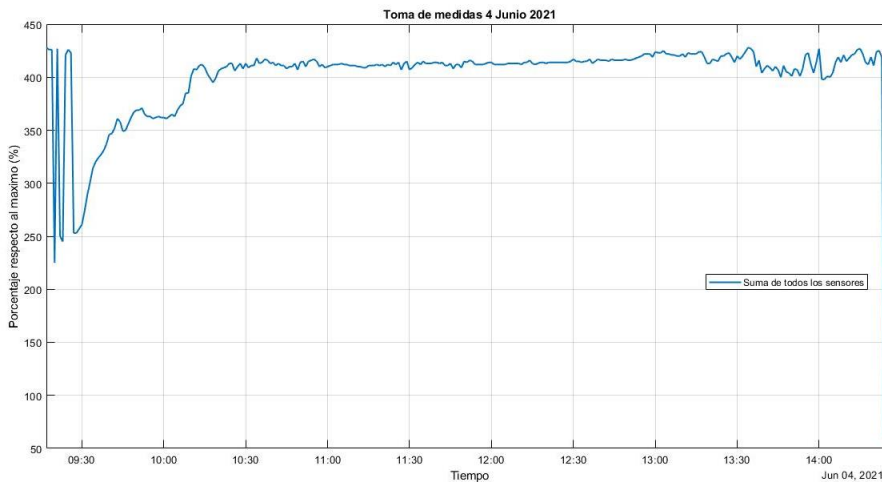


Figura 45. Suma de medidas 4 de junio

Con una media y desviación estándar que valdrá:

Media: 400.8641

Desviación estándar: 38.7036

A pesar de tener las variaciones causadas por las nubes presenta una media bastante alta y una reducida desviación.

Los experimentos siguientes se hicieron con el tubo de 5 cm pues era más ligero, menos propicio a variaciones bruscas por el viento y con una instalación más cómoda de las capas del filtro, además de que se veía que su resultado era similar al de 10 cm.

5.1.6 Medición con tubo de 5 cm y filtro óptico al máximo

Giré las capas del filtro para que bloqueasen la mayor irradiancia posible y lo medí con el flash del móvil, arrojaba valores de 40, y decidí entonces hacer la prueba a lo largo del día para ver su comportamiento, que está representado en la Figura 46:

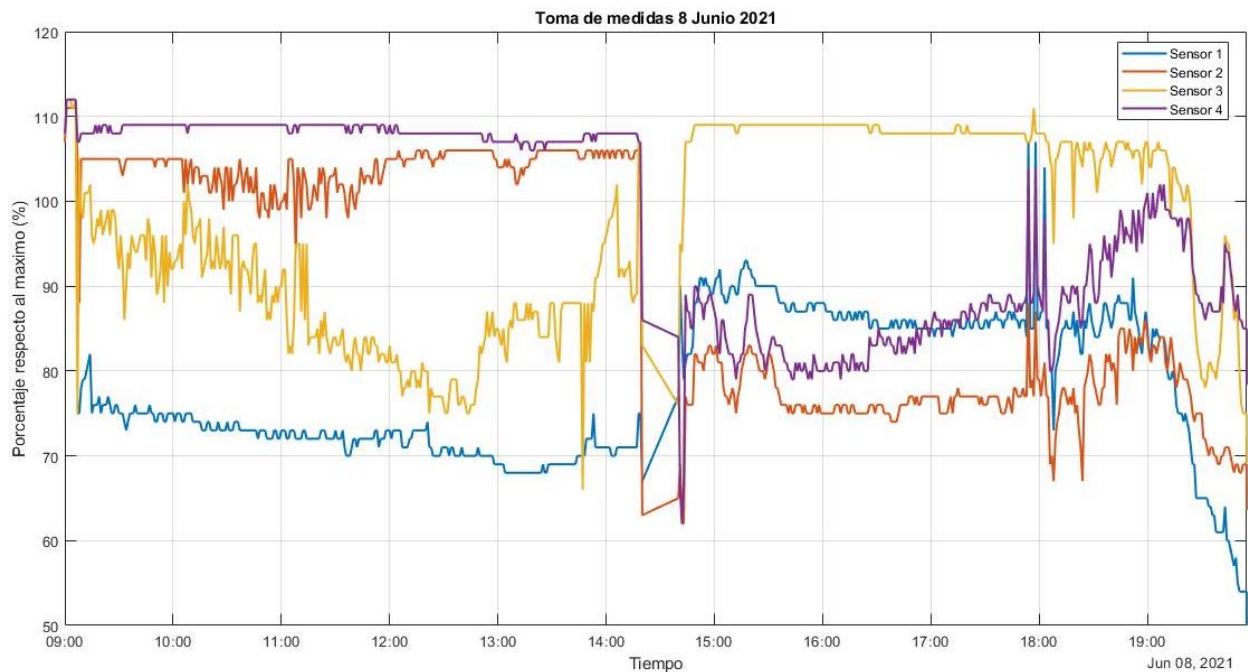


Figura 46. Medidas del 8 de junio

- Al bloquear mayor cantidad de irradiancia, vemos que solo uno satura mientras los demás varían, pues según parece no se encuentra bien orientado al Sol y no todos captan lo mismo, diferente a los casos anteriores en los que eran las medidas más homogéneas.

Esto se tratará de mejorar con la implementación de algoritmos que nos permitirá afinar los datos.

La variación de las sumas de las medidas es algo que podemos ver en la Figura 47:

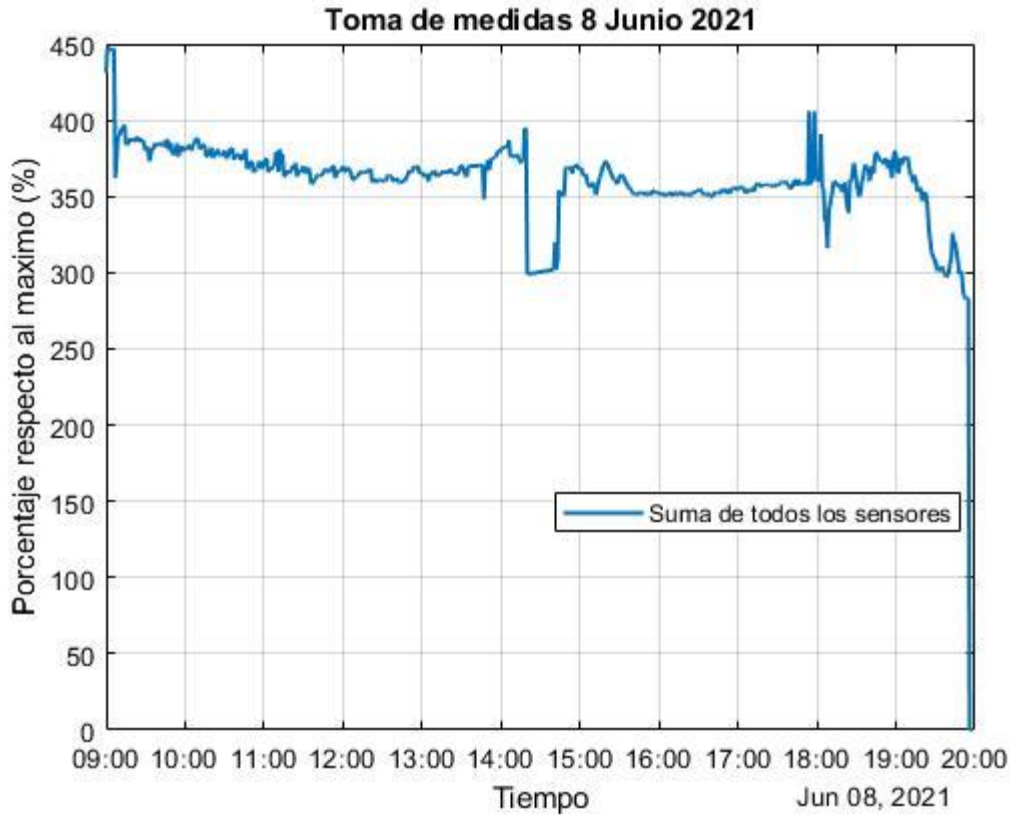


Figura 47. Suma de medidas 8 de junio

Se puede destacar la caída en el periodo de cambio de hora y un comportamiento algo extraño, pues por la mañana decae el valor y a media mañana remonta y por la tarde se mantiene y al final de la tarde se incrementa hasta que decae por no recibir irradiancia.

Por otro lado, en este caso los valores del vector tendrán valores distintos en función de la mañana o el día entero:

Media parcial: 365.7531 Desviación estándar parcial: 19.711

Media: 361.5 Desviación estándar: 31.9331

La media es algo menor por culpa del tramo de por la tarde ya que se ve que se queda entorno 350 y es destacable la diferencia de la desviación estándar siendo la del día completo un 62% mayor que la parcial, probablemente se deba al periodo de 14:20h-14:40h y 19:30h-20:00h que presentan mayor dispersión.

5.1.7 Medición con tubo de 5 cm, filtro óptico y tapón

En vista de los resultados, se decidió hacer otra modificación. Esta consiste en un tapón realizado con cartulina negra y que tiene un agujero en el centro, de manera que cuando se este alineado con el Sol, la luz que llegue haga aumentar los valores de estos y sino tendrán un valor bajo. El resultado se puede ver en la Figura 48:

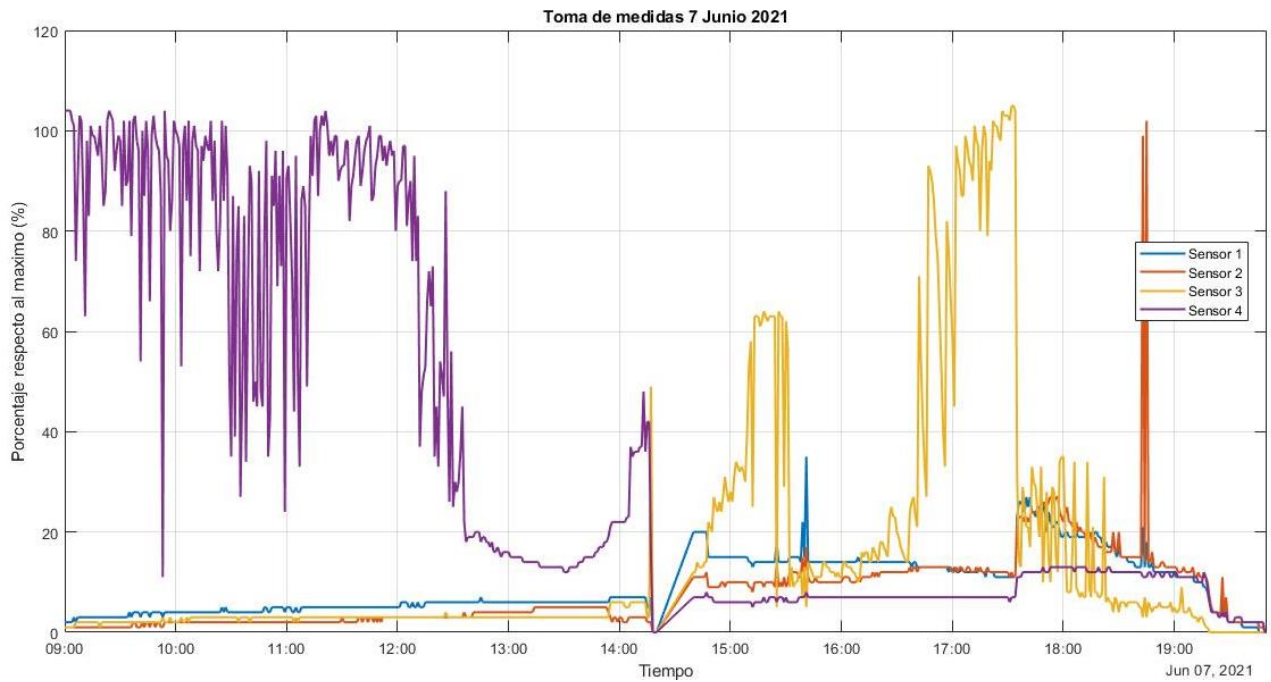


Figura 48. Medidas del 7 de junio

- El comportamiento es el esperado, el sensor al cual le llega la irradiancia a través del agujero llega a saturar, pero no de forma estable pues varía bastante y los demás mantienen un valor constante, aunque con ligeras variaciones, sobre todo por la tarde.

Al mostrar casi todos valores bajos, este modelo si será útil para aplicar los algoritmos pues creo que se podrá apreciar una diferencia, respecto al bucle abierto mostrado.

En la Figura 49 estará sus valores a lo largo del día:

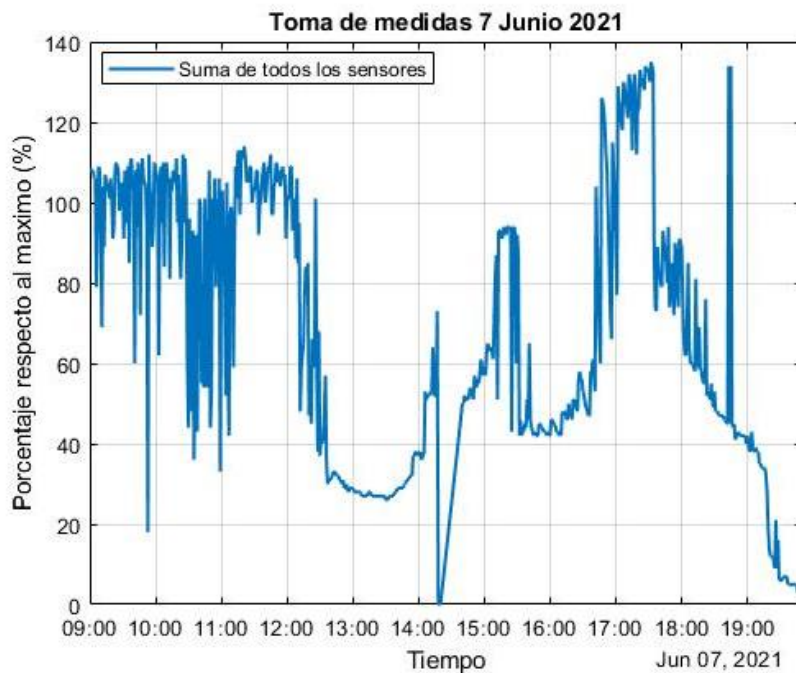


Figura 49. Suma de medidas 7 de junio

Nuevamente aquí dividimos los valores matemáticos en dos tramos, uno parcial que abarca la mañana desde las 9:00h – 14:20h y otro completo que nos arrojan lo siguiente:

Media parcial: 72.4625 Desviación estándar parcial:33.5763

Media: 67.6762

Desviación estándar: 33.9342

Como se ve en la primera parte se ve que la media es más alta y se puede apreciar cualitativamente, pues por la mañana la suma permanece más tiempo en valores cercanos a 100 aunque desde las 12:30h cae bruscamente. Aunque a pesar de tanta variación la desviación en ambos tramos es bastante parecida y menor que en otros experimentos.

Con todos estos experimentos y resultados, hemos acabado el primer bloque del capítulo. Algunas conclusiones previas podrían ser:

- Las LDRs no son los mejores dispositivos para esta medición ya que incluso desde primera hora de la mañana saturan, una alternativa a ellas podría haber sido un fotodiodo
- El filtro no se comporta tan bien como se pensaba, pues, aunque estén en el máximo giro sus capas, no contrarrestan tanta irradiación como parecía, ya que se observan en las imágenes que sigue saturando. Además, no tiene un valor uniforme para cada LDR como en el giro de 30° sino que las capas no tienen una anulación homogénea de la irradiancia.
- La orientación, en general, es buena durante bastante tiempo, aunque por la tarde suele sufrir mayores variaciones. Pienso que esta clase de servos quizás no sean los más adecuados para estos experimentos pues, no son muy precisos.
- Tubos de mayores tamaños, al contrario de lo que pensaba, no anula más irradiancia, pues como se ha visto en las gráficas, no hay una diferencia apreciable por 5cm más.
- Aunque en pruebas con bombillas o flash arrojen valores de 40 o 50, al no tener el flash todas las longitudes de onda, no servirá para hacerse a la idea de los valores verdaderos de irradiancia.

5.2 Algoritmos de mejora.

Los programas que se plantearon para mejorar el comportamiento del dispositivo son tres:

1. Algoritmo basado en puntos aleatorios: en un minuto se comprobarían y se tomarían medidas de diez posiciones cercanas al punto calculado por las ecuaciones de Spencer y se almacenaría su máximo para comprobar la eficacia.

En este caso se tomaría un cuadrado de cuatro grados de lado (centrado en 0, con dos a la derecha y dos a la izquierda). Hay un total de 25 posiciones a elegir, ya que solo se pueden coger números enteros, una representación de este algoritmo se puede observar en la Figura 50.

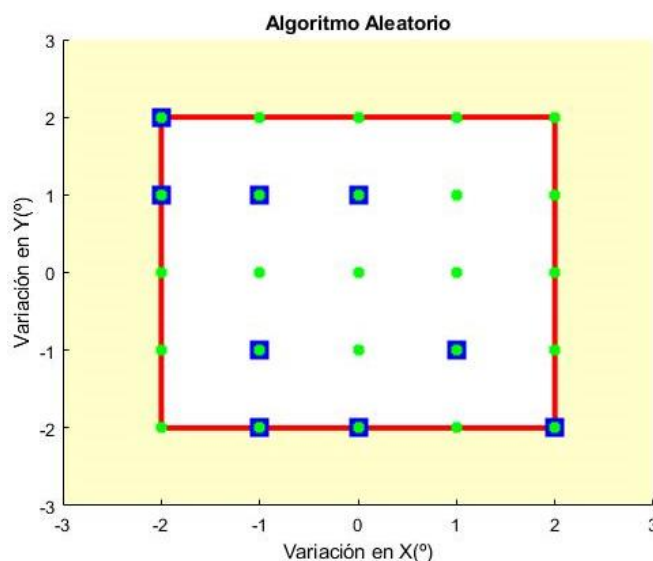


Figura 50. Algoritmo basado en puntos aleatorios.

Los puntos verdes son todas las posiciones para elegir, y los azules serían los cogidos en esta ocasión. Para cada iteración en cada minuto, dichos puntos cambiarán.

- Algoritmo basado en espiral: de forma parecida al anterior en un minuto se tomarán diez medidas, que están situadas en una espiral cuyo radio es también de dos grados. Dichos puntos se compararán con los anteriores y si es mayor se almacenará, sino se descarta. Una imagen de este algoritmo se puede ver en la Figura 51.

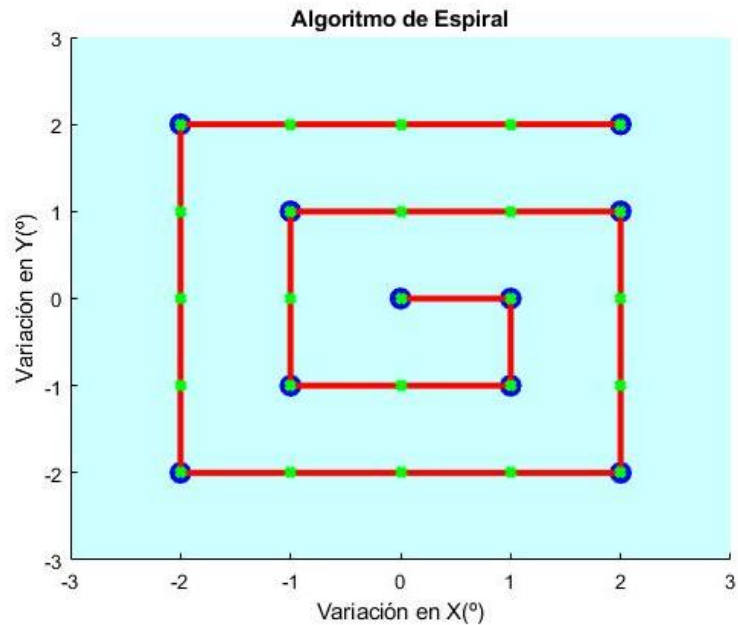


Figura 51. Algoritmo basado en espiral

Aquí, al igual que el anterior, los puntos verdes son todas las posibles combinaciones y los azules son los puntos elegidos para tomar medidas.

- Algoritmo basado en la medida de las LDRs: en este caso se usarán las medidas de las LDRs y unas piezas diseñadas en 3D que le harán sombra si no está bien orientado. Si las medidas de las 4 LDRs son iguales o parecidas con un pequeño margen, se almacenarán y sino es así, querrá decir que a alguna de ellas le llega más irradiación, luego el dispositivo se orientará y girará hasta que todas ellas presenten la misma medida. En la Figura 52 se verá como se ha dividido el soporte de las LDRs:

Algoritmo basado en la medida de las LDRs

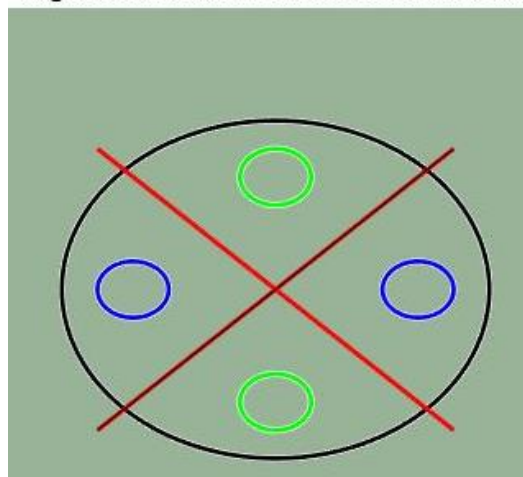


Figura 52. Algoritmo basado en la medida de las LDRs

Se dividirán las 4 LDRs, en dos bloques que permitirán orientarlo horizontal y verticalmente (azul y verde respectivamente), se ira comprobando en cada iteración la diferencia entre las LDRs de cada grupo y si este valor es mayor de un margen se aumentará o disminuirá la altitud o acimut en función del grupo, de esta manera el dispositivo con unas pocas iteraciones será capaz de corregir los posibles errores que tenga.

Una vez descritos los algoritmos que se usarán, se puede empezar con las pruebas y resultados obtenidos. Las pruebas se realizarán con el tubo de 5 cm, salvo que se especifique lo contrario y con las capas del filtro al máximo y otras con el tapón, pues son los dos casos en los que más diferencia se podrán apreciar.

5.3 Experimentos con capas del filtro al máximo y tubo de 5cm.

5.3.1 Medición con algoritmo aleatorio.

Aquí se usó el algoritmo aleatorio ya explicado arriba para realizar pruebas. El resultado se encuentra en la Figura 53.

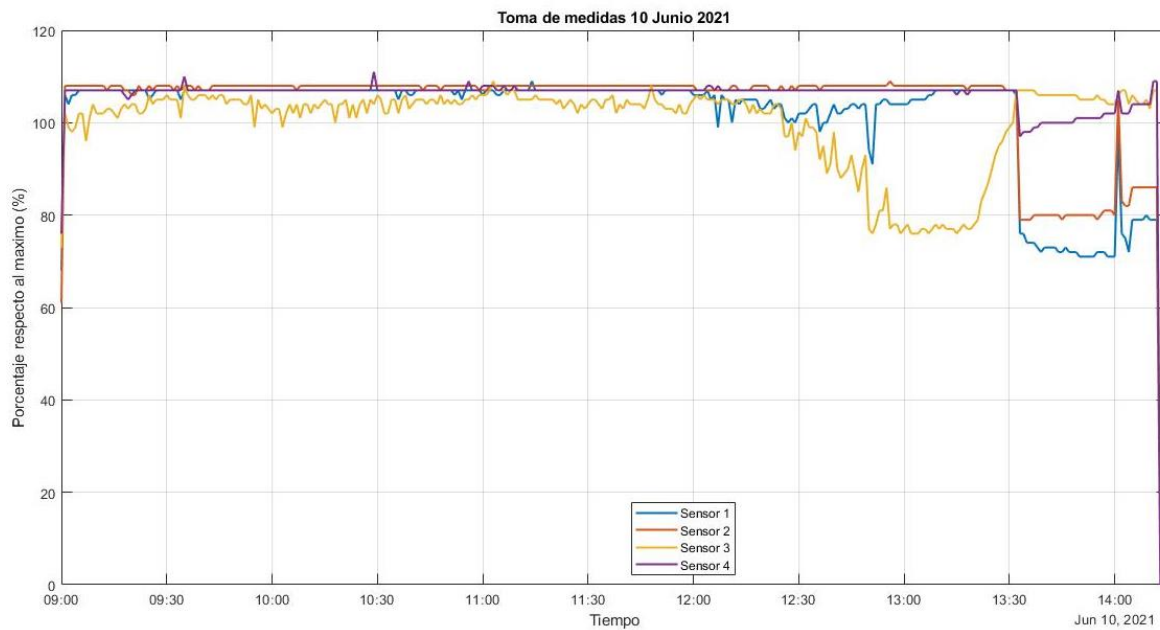


Figura 53. Medidas del 10 de junio

Aunque, se puede apreciar diferencias respecto de los valores tomados días anteriores en bucle abierto en la misma situación, ya que salen más estables. Hay que destacar que saturan también, hay que tener en cuenta que esta mejoría se puede deber a una mejor orientación del dispositivo, ya que una pequeña variación de este puede provocar grandes cambios en los resultados.

En la Figura 54 tenemos las sumas de las medidas de las LDRs:

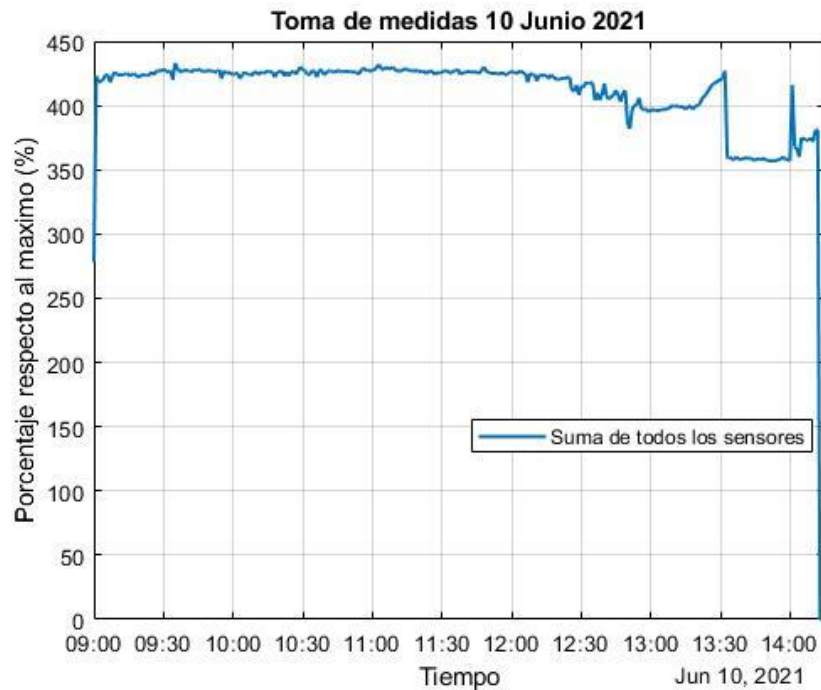


Figura 54. Suma de medidas 10 de junio

Con estos valores, se arrojan las siguientes características:

Media:410.1429

desviacion estándar:39.9786

Se puede ver un valor alto de media comparado con otros y una desviación también alta que probablemente ha sido causada por la caída de valores desde las 12:00h

5.3.2 Medición con algoritmo en espiral.

En la Figura 55, veremos como han sido los resultados durante una mañana ejecutando el algoritmo de espiral.

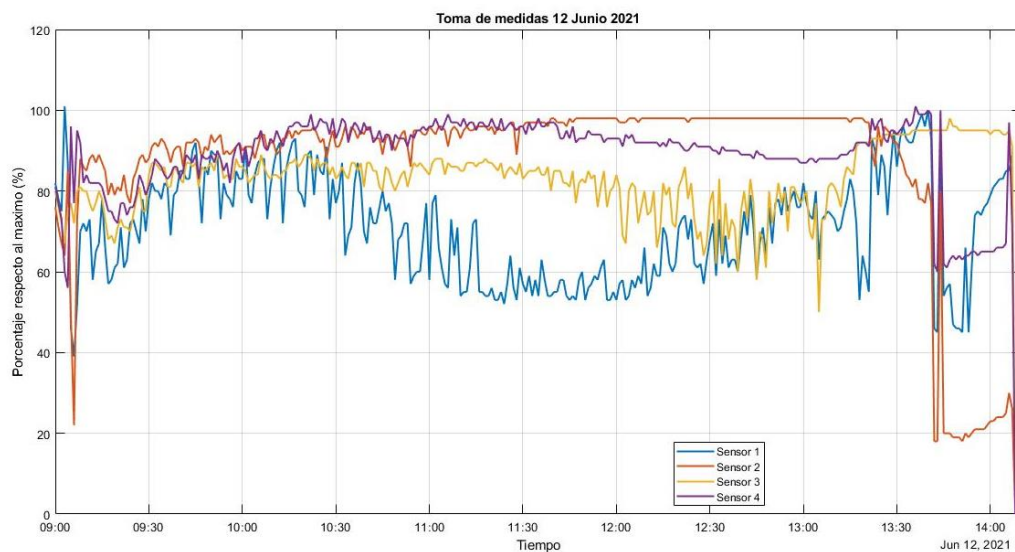


Figura 55. Medidas del 12 de junio.

Vemos que a pesar de las nubes iniciales a las 9:15h que continúan hasta las 10:00h, el comportamiento es parecido al bucle abierto, al final de la mañana se observa que se tiene que orientar mejor y que no difieren tanto los valores.

Con la Figura 56 se mostrará la suma de las medidas en el día 12 de junio:

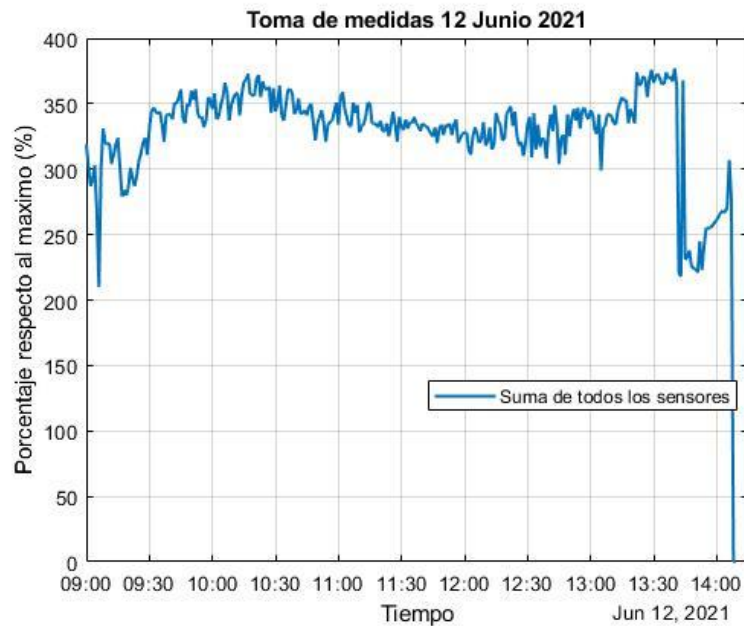


Figura 56. Suma de medidas 12 de junio

A partir de esta gráfica, tenemos los valores siguientes:

Media:326.6677

Desviación estándar:41.3677

La media se observa baja, debido a las nubes iniciales y la caída en medidas de las 13:45h, y una desviación algo alta, normal teniendo en cuenta la variación de valores apreciables en las gráficas.

5.3.3 Medición con algoritmo basado en la medida de las LDRs.

Para este experimento se usarán las paredes diseñadas, impresas y detalladas en el capítulo 3 de este trabajo, se disponen de paredes de 1,2 y 3 cm con las que se verá cual ofrece un mejor comportamiento y cumple el objetivo. En la Figura 57 se puede observar el resultado con la pared de 3cm que era la única con la que se apreciaba una ligera diferencia, pues el filtro ya de por si causaba que los sensores no tuvieran valores homogéneos:

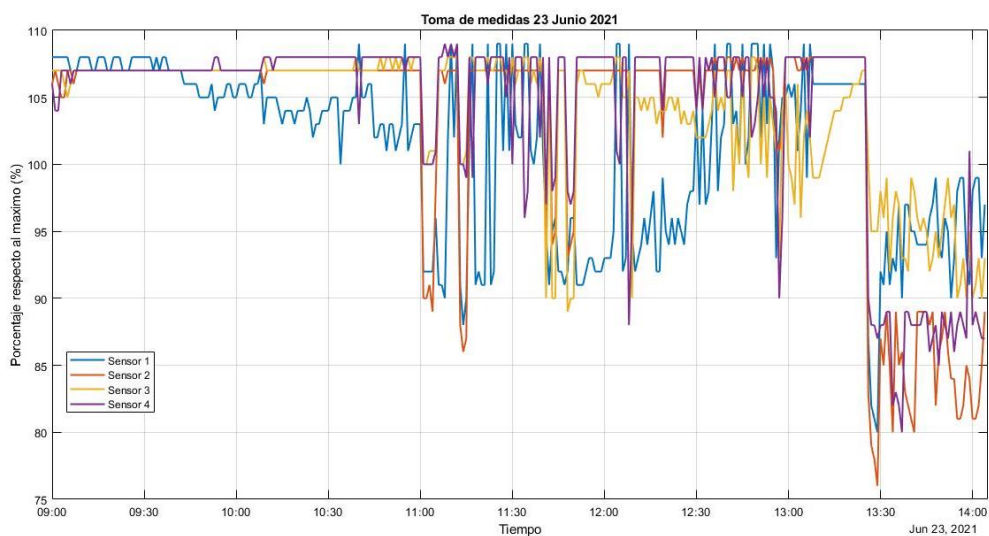


Figura 57. Medidas 23 de junio

Se puede ver como durante las primeras horas de la mañana tienen un valor constante, pero al estar el día a ratos parcialmente nublado se puede ver la disminución de valores entre las 11:00h a 12:00h y el tramo de las 13:30 h hasta el final de la toma de medidas.

Si observamos la suma de medidas de las Figura 58 podremos corroborar esto:

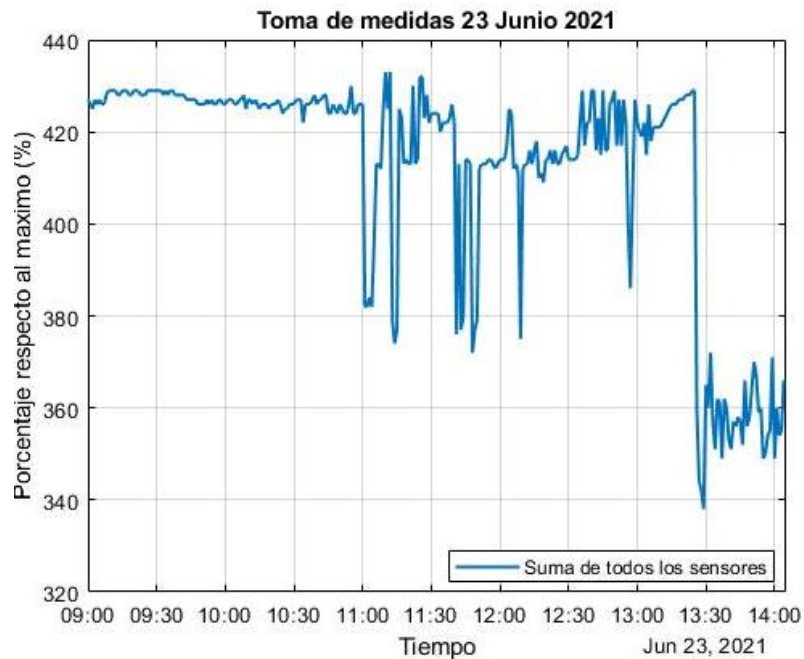


Figura 58. Suma de medidas de 23 de junio

Media:412.7016

Desviación estándar:23.1359

La media arroja un valor algo más bajo a otros casos, pero como se ha comentado principalmente se deberá a la climatología, mientras que la desviación por el contrario tiene el valor más pequeño de todos los experimentos.

Otros datos, que se pueden sacar de este experimento pueden ser, cuales son la altitud aplicada respecto de la calculada y el acimut establecido respecto al calculado, en la Figura 59 lo podemos ver:

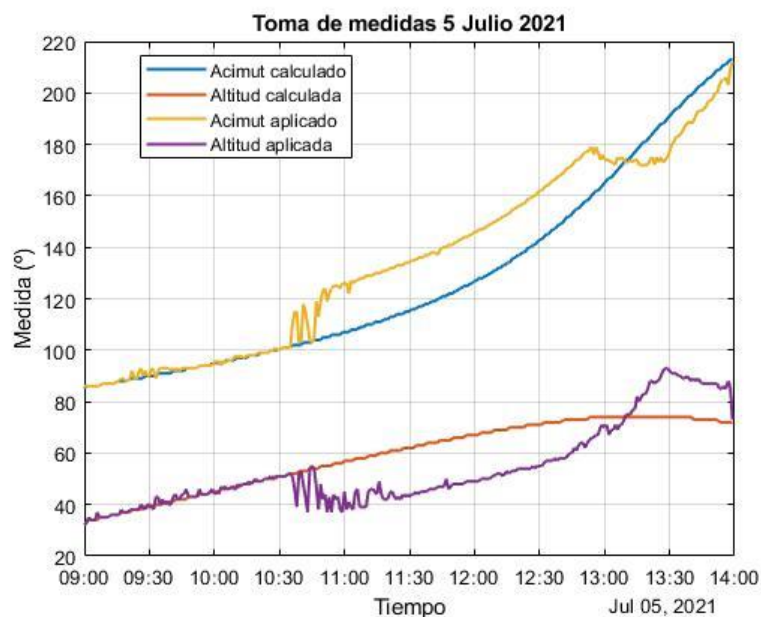


Figura 59. Altitud y acimut 5 de julio

Se puede observar que empiezan bien, pero a partir de las 10:30h ambos siguen un comportamiento distinto, el acimut se incrementa y la altitud, al contrario, mientras que, estando el Sol en su posición más alta a final de la mañana, se dan la vuelta los valores. Si observamos la Figura 60, tenemos la diferencia de los valores.

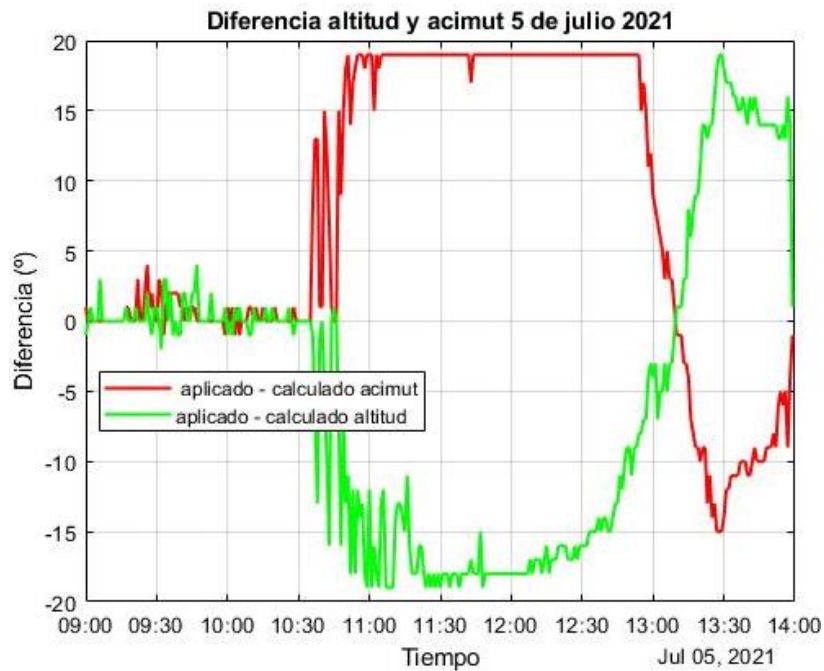


Figura 60. Diferencia de valores 5 de julio

Tal y como se ha comentado, tienen comportamientos contrarios como se ve, al desplazarse uno vemos que influye al otro, también es destacable comentar dos aspectos importantes:

En primer lugar, los valores alcanzados son del orden de entre 15° y 20° , en valor absoluto, luego con los algoritmos calculados y ejecutados anteriormente no llegaríamos, la idea de ampliar su rango se ve fortalecida gracias a este experimento.

Después, al no poder disponer de los valores correspondientes de los sensores, no se puede saber si estos muestran un comportamiento similar a la Figura 57 o son diferentes, ya que el filtro no satura de la misma manera y puede que se generen disparidades en las medidas que hagan que este se desplace en un sentido u otro. Para hacer pruebas de 5h el Arduino solo puede recoger datos en una tabla de 4 columnas, si aumentamos en una esta, el tiempo se divide por la mitad y así sucesivamente.

Tras haber analizado los experimentos con los diferentes algoritmos, se debe de comentar que es difícil poder concluir algo con claridad. Pues al saturar las LDRs, las comparaciones son difíciles y al variar las medidas de forma drástica por la variación de unos grados la altitud y el acimut, es posible que el dispositivo no estuviese bien orientado en los primeros experimentos. Luego, puede arrojar valores que no se corresponden con lo que en realidad es, aunque para eso se hicieron varias pruebas y como se ve, cada una tiene una media y una desviación. Un punto destacable si puede ser, que mediante el algoritmo basado en la medida de las LDRs la desviación estándar anota un valor bastante más bajo que en cualquier otro caso, por lo que las medidas ofrecen menos variación y nos permite comprobar que el posicionamiento del dispositivo puede llegar a cambiar bastante.

5.4 Experimentos con tapón y tubo de 5cm.

5.4.1 Medición con algoritmo aleatorio.

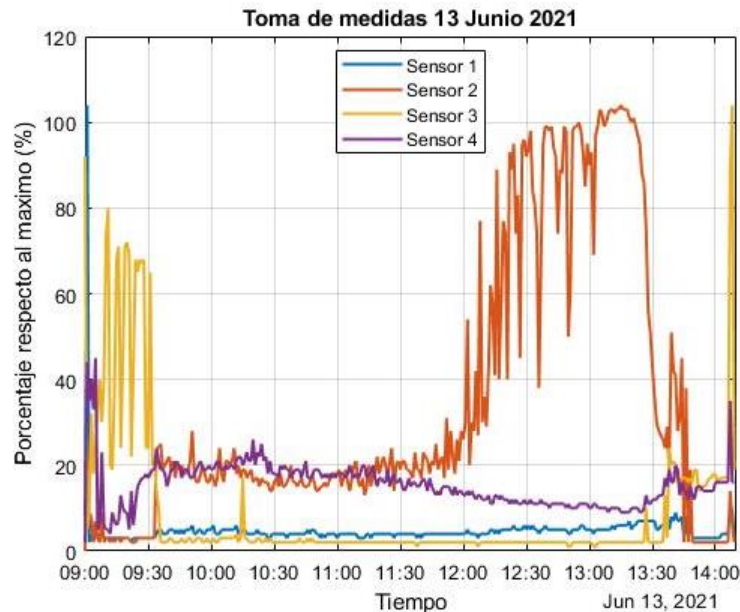


Figura 61. Medidas 13 de junio

Analizando la Figura 61, se puede observar discrepancias respecto a la inicial con el tapón en bucle abierto, los valores de los sensores son en general un poco más altos. Mientras la Figura 62 nos permite comprobar que la hipótesis no era la adecuada:

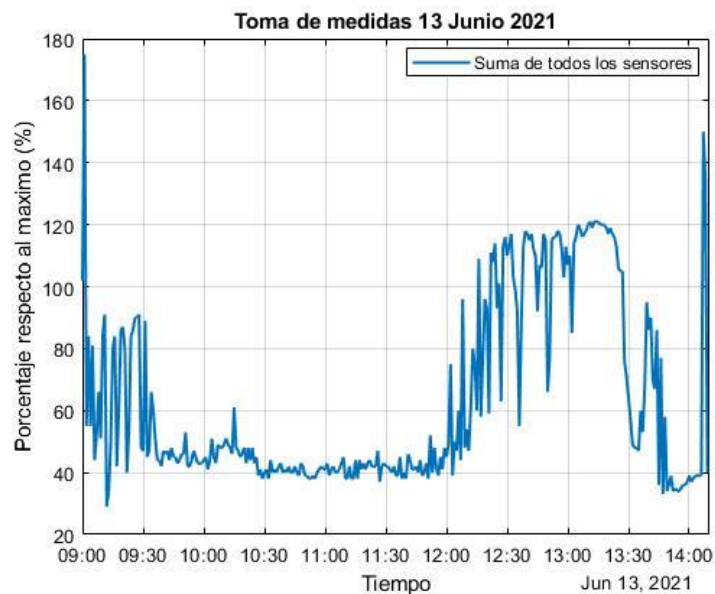


Figura 62. Suma de medidas 13 de junio.

Con los valores de:

Media: 63.78

Desviación estándar: 30.18

Vemos que la media es algo inferior al experimento en bucle abierto, pero hay que tener en cuenta, que en este caso las medidas son muy variables, pues al pasar la irradiancia solo por un agujero hace falta una orientación bastante precisa y puede que el rango del algoritmo sea pequeño. Haría falta, que en lugar de coger medidas en

un rango de -2 a 2 respecto al acimut o altitud fuesen el doble o el triple para poder notar diferencias, pues la variación de 1° la altitud, en algunos casos no provoca ni variación en el servo y otras varía solo en 1. Y un incremento tan pequeño, en un dispositivo con una amplitud de 180° no cambia prácticamente nada.

5.4.2 Medición con algoritmo en espiral.

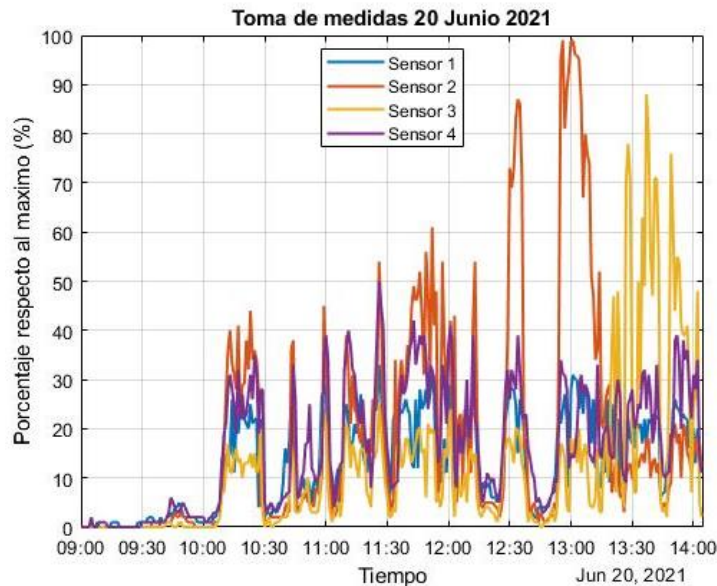


Figura 63. Medidas 20 de junio

Observando la Figura 63, primero habría que destacar que el día 20 de junio, fue un día bastante nublado y se observa al inicio que, durante una hora, ningún sensor fue capaz de tomar una medida superior a 10. Además, hay un comportamiento de estos que perdura hasta el final de la mañana, mientras que, en los demás casos, usando el tapón normalmente destaca uno de ellos y los demás permanecen con un valor bajo, aquí salvo al final de la mañana crecen todos de manera uniforme.

Una posible razón es que al estar parcialmente nublado la luz entra de manera más homogénea y tienen todos valores parecidos, pues al final si presenta un comportamiento parecido al anterior experimento, un sensor con un gran valor y los demás con valores bajos.

Si analizamos la Figura 64 podemos ver un comportamiento también algo anómalo, ya que sus datos serán los siguientes:

Media: 60.66

Desviación estándar: 50.0363

La media resulta más alta de lo que podría esperarse para un día como el que hizo, ya que solo es tres puntos inferior a la del espiral y seis inferior al primer experimento. La desviación si que es bastante alta ya que hay una gran variación entre valores en unos minutos vale 120 en los próximos 20 y eso hace incrementar la desviación.

Considero que en este caso, si no hubiese sido por las nubes el experimento hubiese podido obtener unos resultados mejores a los iniciales, pues con un leve aumento de los valores en la primera hora de la mañana hubiese propiciado un aumento en la media y disminución de la desviación estándar. No obstante, el resultado es satisfactorio, al poder haber comprobado el comportamiento del dispositivo ante esta situación climatológica.

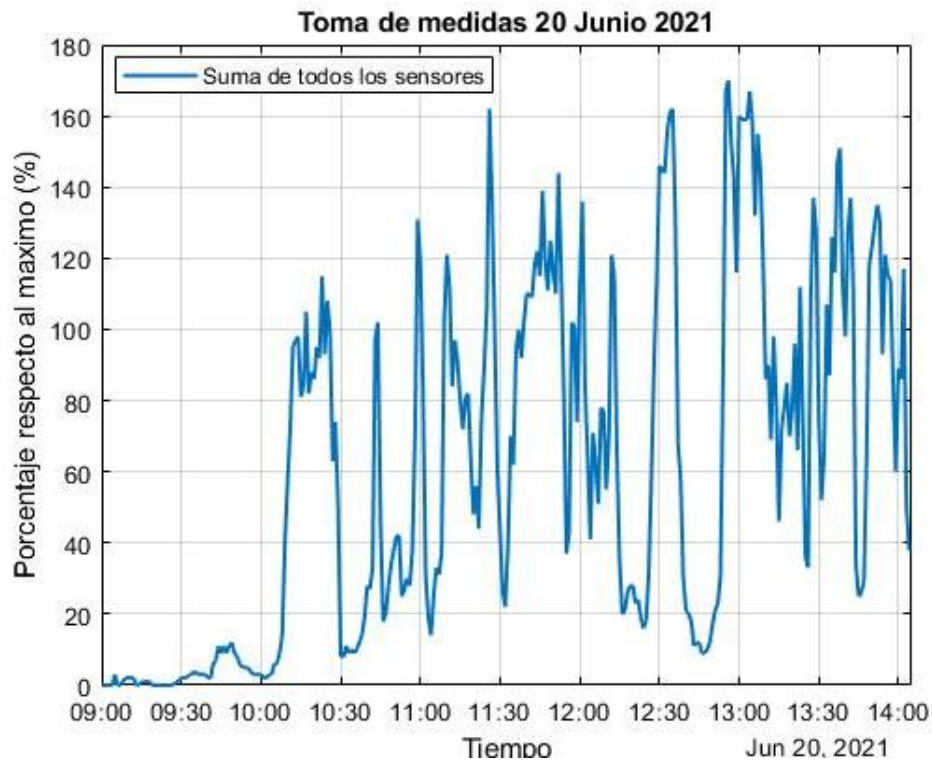


Figura 64. Suma de medidas 20 de junio

5.4.3 Medición con algoritmo basado en la medida de las LDRs.

En la Figura 65 veremos los resultados:

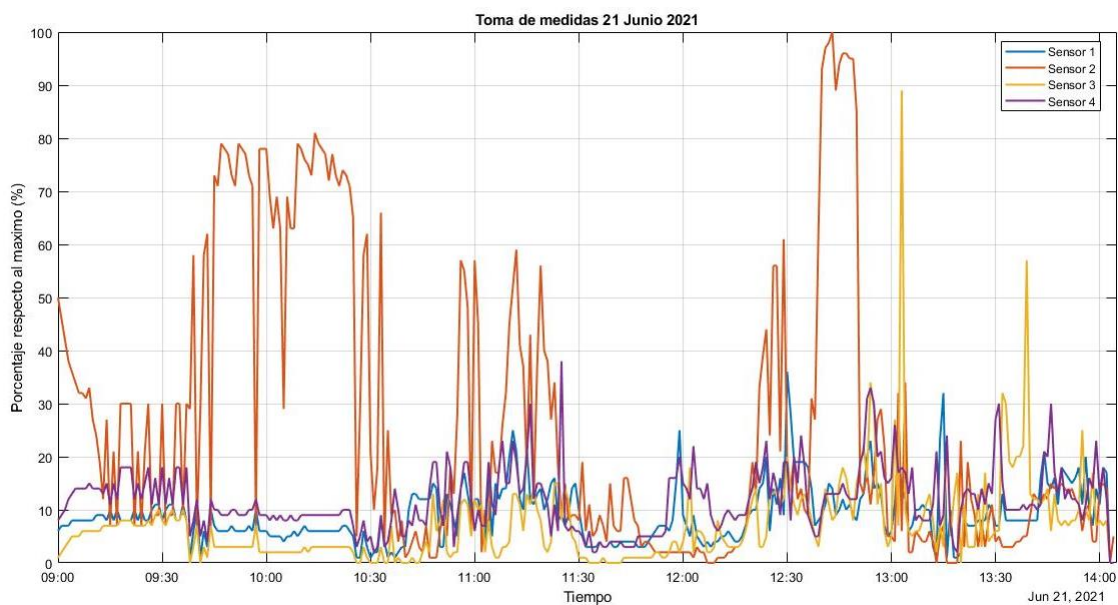


Figura 65. Medidas 21 de junio

Analizandola se puede ver que, aunque hay momentos en los que un sensor sobresale y casi siempre es el 2, los demás valores oscilan y en varios puntos de la mañana tienen valores similares ya que al distribuir la luz y

no llegarle solo a un sensor estos son algo más altos a costa de reducir al que si le llegue mas irradiancia, en la Figura 66 veremos la suma y después los valores obtenidos:

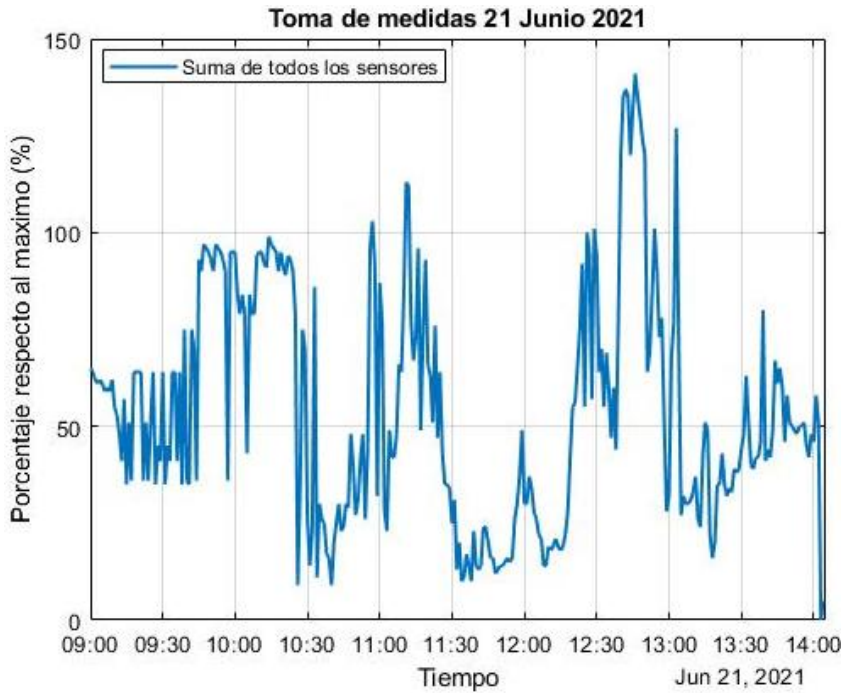


Figura 66. Suma de medidas 21 de junio

Media: 52.334

Desviación estándar: 30.2515

Se puede observar un comportamiento más homogéneo de la suma, presenta variaciones, pero no son tan bruscas como en otros experimentos. Esto se puede apreciar en los valores, con una media más baja como consecuencia de una mayor uniformidad, así como de una desviación estándar mas baja que en en otros algoritmos o al mismo nivel.

En la Figura 67, tenemos como es el comportamiento del acimut y de la altitud a lo largo de la mañana y veremos la diferencia que hay entre las calculadas y las que ejecuta al final el dispositivo:

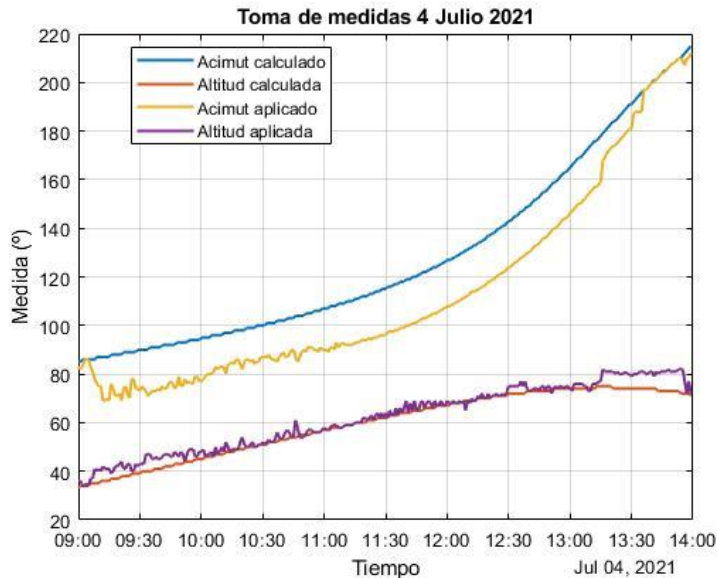


Figura 67. Altitud y acimut 4 de julio

Antes de analizar, se observará la Figura 68 en la cual está la disconformidad de las medidas respecto al 0:

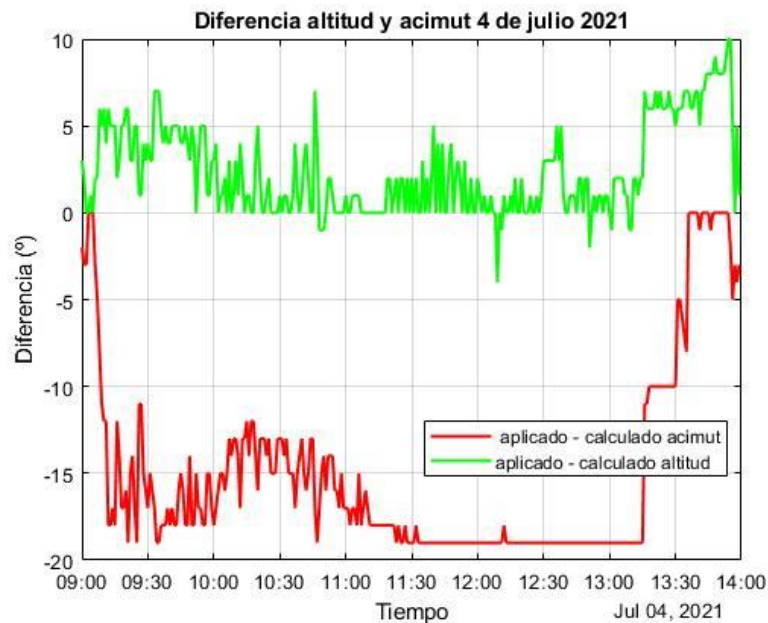


Figura 68. Diferencia de valores 4 de julio.

En la altitud el cambio es poco apreciable en las primeras horas de la mañana, mientras que a final de esta y con el Sol situado a una mayor altura si que alcanza un pico de 10° . Mientras, el acimut se comporta al revés en todo momento, es inferior el que se acaba posicionando al que se calcula y al final de la prueba llega irradiancia más uniforme y reduce la diferencia. También, hay que destacar que como se vio en las gráficas correspondientes a las medidas de los sensores, este tipo de experimento no presenta valores homogéneos y estará todo el tiempo moviéndose, hasta que los valores sean parecidos, cosa que casi nunca se cumple, ya se ve en la Figura 65, luego en este caso son unas gráficas más orientativas y sirve para ilustrar que con los otros algoritmos tampoco llegaría nunca a cumplirse la homogeneidad en las medidas de los sensores.

En esta ocasión, tras finalizar estos resultados, el que ofrece unos valores más distintos, es también el algoritmo basado en la medida de las LDRs ya que como se ha comentado, las medidas son más uniformes y se aprecia en su media, más baja y su desviación estándar. Para el de espiral al haber estado el día nublado la media desciende levemente y la desviación aumenta, mientras que en aleatorio es como el experimento sin algoritmo. Aquí también hay que recalcar que una leve desorientación del dispositivo provoca un gran cambio pues al pasar la irradiancia solo por un pequeño agujero si varía su situación las medidas se verán gravemente alteradas.

6 CONCLUSIONES

6.1 Conclusiones.

De este trabajo se pueden sacar conclusiones de diversos aspectos, las más destacadas pueden ser:

- Se han calculado las ecuaciones solares para obtener el acimut y altitud.
- Se ha programado el Arduino para seguir colocarse en función del resultado de las ecuaciones.
- Se han controlado diversos sensores y actuadores ya sea la brújula, el regulador LM2596, LDR
- Se ha diseñado un soporte para las LDRs y para el pan and tilt que ha cambiado en varias ocasiones.
- Se ha diseñado y desarrollado un circuito de conexiones.
- Se ha realizado nuevos tubos para poder realizar experimentos.
- Se han probado diferentes formas del filtro para poder ver su eficacia.
- Se usan distintos tubos para ver como varía la toma de medidas.
- Se han desarrollado e implementado diferentes algoritmos de mejora de toma de datos.

6.2 Futuras mejoras.

Durante la realización de este trabajo se han ido observando algunos problemas que han ido surgiendo y se verá como poder solucionarlos como posibles mejoras futuras:

- Las LDRs no son adecuadas para este proyecto. Actualmente hay otros dispositivos que nos proporcionan una medida de la irradiancia incidente como son los fotodiodos o incluso las propias celdas fotovoltaicas.
- Ampliar el rango de los dos primeros algoritmos y como consecuencia, tomar más muestras para mantener una ratio de medidas/posibilidades alto.
- Usar otro microcontrolador, que posea mejores funciones y posibilidades, más memoria, velocidad de reloj, con convertidores ADC más precisos; los dos que podrían ser más adecuados para este trabajo pueden ser:

-ATTINY3226: microcontrolador con 32kb de flash, 3kb de SRAM y 256b de EEPROM, poseen un tamaño reducido, este en concreto tiene 20 pines, ADC de 12 bits en lugar de 10 del Arduino, diferentes referencias de tensión de entrada, UART, SPI, reloj de tiempo real incorporado (RTC) y con un precio de 1€. Su programación se haría también mediante el Arduino IDE al cual habría que añadir una librería. En la Figura 61 podemos ver el Attiny1626:

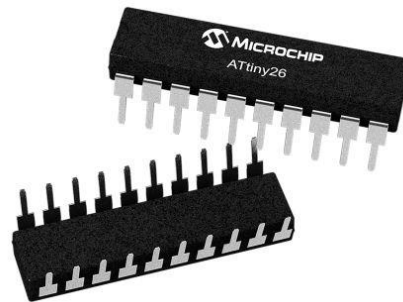


Figura 69. Attiny1626.

-ESP32: Microprocesador de 32bit de doble núcleo que puede funcionar a 160MHz o 240MHz, 520kiB de SRAM, Wi-Fi 802.11 b/g/n, Bluetooth v4.2, modo de bajo consumo, 36 pines, ADC de 12 bits y DAC 8 bits, hasta 16 PWM, igual que el anterior, mediante un complemento puede ser programado con Arduino IDE, en la Figura 62 estará en el micro:



Figura 70. ESP32

- Usar un pan a tilt con más robustez y así, el acimut no dependería del eje de un microservo, en el mercado hay varios pan and tilt que solucionan este problema, poseen posiciones diferentes para evitar esto.
- El micro servo empleado para la altitud es un Micro Servo SG90, posee un torque de 2.3 kg·cm, puede levantar un peso como máximo de 2.3kg a un centímetro de distancia. Hay que recordar que el tubo de 10 cm pesaba 24g y que había más tubos de 15,20,25,30 cm, luego el servo parece que no es el adecuado para esto. Una solución podría haber sido usar servos de un mayor tamaño y metálicos, así el eje sería de mayor.
- Una posible mejora más hubiese sido, implementar una alimentación a base de una batería que sería cargada por la placa, para ello habría que haber rediseñado la caja para poder albergarla.
- El filtro a pesar de mostrar que funciona; no filtra tanto como debería, ya que como se ven en los experimentos, sigue saturando. Se podría haber usado otro filtro polarizado para obtener mejores resultados.

7 REFERENCIAS

- [1] Mestre Barceló, A. (1994). ¿Cómo se mide la radiación solar? *Revista de meteorología*, p. 15-18.
- [2] Planas, O. (4 de Febrero de 2019). *Solar Energia*. Obtenido de <https://solar-energia.net/que-es-energia-solar/radiacion-solar/irradiacion-solar>
- [3] Korevaar, M. (20 de Abril de 2015). *KippZonen*. Obtenido de https://www.kippzonen.es/News/596/La-importancia-de-la-respuesta-a-la-temperatura-del-piranometro#.YMu_IGzblX
- [4] Arsus Energia S.L. (13 de Noviembre de 2020). *Arsus Energia*. Obtenido de <https://arsusenergia.es/centrales-termosolares-su-funcionamiento-y-tipos/>
- [5] Fritzling. (12 de Mayo de 2021). *About/Context*. Obtenido de <https://fritzling.org/about/context>
- [6] FreeCAD. (s.f.). *FreeCAD web org*. Obtenido de https://www.freecadweb.org/downloads.php?lang=es_ES
- [7] SolidWorks. (s.f.). Obtenido de <https://www.solidworks.com/es/contact-us>
- [8] Ultimaker. (s.f.). Obtenido de <https://ultimaker.com/es/about-ultimaker>
- [9] García, A. F. (2016). *Posición del Sol en el Sistema de Referencia Local*. Obtenido de <http://www.sc.ehu.es/sbweb/fisica3/celeste/sol/sol.html>
- [10] Garcia, M. S. (Septiembre de 2009). *Control y optimización del seguimiento solar con dos grados de libertad para mejorar el aprovechamiento energetico*. Madrid, Comunidad de Madrid, España.

ANEXO A: Código “bucle abierto”

```
#include <Servo.h>
#include <Wire.h>
#include <MechaQMC5883.h>
#include <TimeLib.h>

time_t t0,t1,dif;
Servo servo_acimut;
Servo servo_altitud;

int tabla[]={31,29,31,30,31,30,31,31,30,31,30,31};
int dias=0;
//float latitud=37.395218*(PI/180); SEVILLA
//float longitud=-5.9797;

float latitud=38.013809*(PI/180);
float longitud=-3.381430;
float declinacion=-1;
int movimiento, alt;
float altitud=0;
float acimut=0;
int minu=40;
int hor=14+1;
int numero;
float maximo=116500;
byte matriz[332][4];
int j=1;

int offset=30;
int fin=0;
byte comprobar=0;
unsigned long periodo=60000;
unsigned long Tiempoahora=0;

const int sensor1=A1;
const int sensor2=A2;
const int sensor3=A3;
const int sensor4=A4;

int valor1,valor2,valor3,valor4;

MechaQMC5883 qmc;

void Servos(int angulo){
  if(acimut<= 180){
    if(angulo>acimut){
      if(angulo-acimut<180){
        movimiento = angulo- acimut;
        if(movimiento>165) movimiento=165;
        else if(movimiento<15) movimiento=15;
        servo_acimut.write(movimiento);
        alt=altitud;
      }
    }
  }
}
```

```

servo_altitud.write(alt); }
else{
  movimiento = angulo - (acimut+180);
  if(movimiento>165) movimiento=165;
  else if(movimiento<15) movimiento=15;
  servo_acimut.write(movimiento);
  alt=180- altitud;
  servo_altitud.write(alt); }
}
else{
  movimiento = angulo - acimut+180;
  if(movimiento>165) movimiento=165;
  else if(movimiento<15) movimiento=15;
  servo_acimut.write(movimiento);
  alt=180- altitud;
  servo_altitud.write(alt); }
}
else{
  if(angulo>acimut){
    movimiento = angulo- acimut;
    if(movimiento>165) movimiento=165;
    else if(movimiento<15) movimiento=15;
    servo_acimut.write(movimiento);
    alt=altitud;
    servo_altitud.write(alt);}
  else{
    if(acimut-angulo>180){
      movimiento = angulo- acimut+360;
      if(movimiento>165) movimiento=165;
      else if(movimiento<15) movimiento=15;
      servo_acimut.write(movimiento);
      alt=altitud;
      servo_altitud.write(alt);
    }
    else{
      movimiento = angulo- (acimut-180);
      if(movimiento>165) movimiento=165;
      else if(movimiento<15) movimiento=15;
      servo_acimut.write(movimiento);
      alt=180-altitud;
      servo_altitud.write(alt);
    }
  }
}
}
}

void setup() {

  // put your setup code here, to run once:
  Serial.begin(9600);

  servo_acimut.attach(6); //pin con pwm
  servo_acimut.write(90);
  servo_altitud.attach(10); //pin con pwm
  servo_altitud.write(30);
}

```



```

void loop() {
  // put your main code here, to run repeatedly:
  if(millis()> Tiempoahora + periodo){
    Tiempoahora=millis();
    t0 = ponFecha(2021, 1, 1, 0, 0, 0);
    t1= ponFecha(2021, 6,9, hor, minu, 0);

    Wire.begin();
    qmc.init();

    dif = t1 - t0;
    float minutos= minute(dif) ;
    float resta= minutos/60;
    float horas=hour(dif)+resta;

    float diferenciapos= abs(longitud*4);

    //Diferencia por horario de verano
    int diferenciames;
    if(month(t1)<4 | month(t1)>10){
      diferenciames=60;
    }
    else{
      diferenciames=120;
    }

    //Ecuación del tiempo
    //Calculamos el angulo diario
    if(year(t1)%4==0){ //año bisiesto
      tabla[1]=29;
    }
    else{
      tabla[1]=28;
    }
    int suma=0;

    for(int i=0; i<(month(dif)-1);i++){
      suma=suma+tabla[i];
    }
    dias= suma+ day(dif);
    float angulodiario=(2*PI/365)*(dias-1);
    declinacion=0.006918-0.399912*cos(angulodiario)+0.070257*sin(angulodiario)-0.006758*cos(2*angulodiario)+
    ...0.0009907*sin(2*angulodiario)-0.002697*cos(3*angulodiario)+0.00148*sin(3*angulodiario);
    float dec=declinacion*180/PI;

    //ecuacion del tiempo
    float et=(0.000075+0.001868*cos(angulodiario)-0.032077*sin(angulodiario)
    ...-0.014615*cos(2*angulodiario)-0.04089*sin(2*angulodiario))*229.18;
    float diferenciatal=diferenciames+diferenciapos-et;
    float Ts=horas-(diferenciatotal/60);

    //angulo horario
    float w=((Ts-12)*15)*(PI/180);

```

```

//altitud
altitud=asin(cos(latitud)*cos(declinacion)*cos(w)+sin(declinacion)*sin(latitud));
//acimut
acimut=acos((cos(latitud)*sin(declinacion)-cos(w)*sin(latitud)*cos(declinacion))/cos(altitud));

altitud=altitud*(180/PI);
acimut=acimut*(180/PI);
if(w>0) acimut=360-acimut;

int x,y,z;
qmc.read(&x,&y,&z);
//Deberemos calcular el angulo del eje x respecto del Norte
float angulo=atan2(y,x);
angulo=angulo * RAD_TO_DEG;
angulo=angulo-declinacion;

if(angulo<0) angulo=angulo +360;

Servos(angulo);

Serial.print(String(horas) + " hora ");
Serial.print(String(diferenciapos) + "dif posicion ");
Serial.print(String(altitud)+ " Altitud ");
Serial.print(String(acimut)+ " Acimut ");
Serial.print(String(alt)+ " Serv altit "+ String(movimiento)+ " Serv acim ");
Serial.println(String(fin)+ " " );

valor1=analogRead(sensor1);
valor2=analogRead(sensor2);
valor3=analogRead(sensor3);
valor4=analogRead(sensor4);

valor1=(float)lumenes(valor1);
valor2=(float)lumenes(valor2);
valor3=(float)lumenes(valor3);
valor4=(float)lumenes(valor4);

matriz[j][0]=valor1;
matriz[j][1]=valor2;
matriz[j][2]=valor3;
matriz[j][3]=valor4;
j++;

minu=minu+1;
fin=fin+1;

if (minu>=60){
  hor=hor+1;
  minu=0;
}

for(int fil=2;fil<=fin;fil++){
  //Serial.print(String(fin)+ " fin ");
  Serial.print(String(matriz[fil][0])+ " ");
  Serial.print(String(matriz[fil][1])+ " ");
  Serial.print(String(matriz[fil][2])+ " ");
  Serial.println(String(matriz[fil][3])+ " ");
}
}
}

```

```

float lumenes(float sensor ){
    float voltaje= sensor*5/1023;
    float resistencia =(10*5/voltaje)+10;
    float lumen =pow(10, (2.29-log10(resistencia))/0.193);
    numero=((lumen*100)/maximo);
    return numero;
}

time_t ponFecha(int y, int m, int d, int hh, int mm, int ss){
    tmElements_t f;
    f.Second = ss;
    f.Minute = mm;
    f.Hour = hh;
    f.Day = d;
    f.Month = m;
    f.Year = y - 1970 ;
    return makeTime(f);    //Crea tiempo Unix
}

```

ANEXO B: Código “aleatorio”

```

float acimut=0;
int minu=30;
int hor=14+1;
int numero;
float maximo=116500;
byte matriz[316][4]={0};
int j=1;
int fin=0;
int maximo1,maximo2,maximo3,maximo4=0;
int r,k;
byte aux=0;
byte variable=1;
unsigned long periodo=6000;
unsigned long Tiempoahora=0;

const int sensor1=A1;
const int sensor2=A2;
const int sensor3=A3;
const int sensor4=A4;

int valor1,valor2,valor3,valor4;

MechaQMC5883 qmc;

```

```

if(angulo<0) angulo=angulo +360;
//tocará mover los servos

    r = random(-2,2);
    k = random(-2,2);

    if(aux>=1){
        altitud= altitud + r;
        acimut= acimut + k;
    }

Servos(angulo);

aux++;
variable++;

valor1=analogRead(sensor1);
valor2=analogRead(sensor2);
valor3=analogRead(sensor3);
valor4=analogRead(sensor4);

valor1=(float)lumenes(valor1);
valor2=(float)lumenes(valor2);
valor3=(float)lumenes(valor3);
valor4=(float)lumenes(valor4);

if(valor1> maximol){
    matriz[j][0]=valor1;
    maximol=valor1;
}
if (valor2> maximo2){
    matriz[j][1]=valor2;
    maximo2=valor2;
}
if (valor3> maximo3){
    matriz[j][2]=valor3;
    maximo3=valor3;
}
if (valor4> maximo4){
    matriz[j][3]=valor4;
    maximo4=valor4;
}

if (variable>10){ //habra pasado un minuto
Serial.print(String(horas) + " hora ");
Serial.print(String(diferenciapos) + "dif posicion ");
Serial.print(String(altitud)+ " Altitud ");
Serial.print(String(acimut)+ " Acimut ");
Serial.println(String(maximol)+ " " + String(maximo2)+ " " + String(maximo3)+ " " +String(maximo4)+ " " );
Serial.print(String(alt)+ " Serv altit " + String(movimiento) + " Serv acim ");
Serial.println(String(fin)+ " " );
minu=minu+1;
}

```

```

minu=minu+1;
j++;
fin=fin+1;
aux=0;
variable=1;
maximo1=0;
maximo2=0;
maximo3=0;
maximo4=0;

if (minu>=60){
  hor=hor+1;
  minu=0;
  }
  for(int fil=2;fil<=fin;fil++){
    Serial.print(String(matriz[fil][0])+ " "); //EMPIEZA EN 1 SI O SI
    Serial.print(String(matriz[fil][1])+ " "); //FILAS EN 1 COLUMNAS DE 0
    Serial.print(String(matriz[fil][2])+ " ");
    Serial.println(String(matriz[fil][3])+ " ");
  }
}
}
}

```

ANEXO C: Código “espiral”

```

time_t t0,t1,dif;
Servo servo_acimut;
Servo servo_altitud;

int tabla[]={31,29,31,30,31,30,31,31,30,31,30,31};
int dias=0;
//float latitud=37.395218*(PI/180);
//float longitud=-5.9797;
float latitud=38.013809*(PI/180); //UBEDA
float longitud=-3.381430;
float declinacion=-1;
int movimiento, alt;
float altitud=0;
float acimut=0;
int minu=10;
int hor=9+1;
int numero;
int servo1[10]={0,1,1,-1,-1,2,2,-2,-2,2};
int servo2[10]={0,0,-1,-1,1,1,-2,-2,2,2};

```

```

if(angulo<0) angulo=angulo +360;
//tocará mover los servos

if(i<10){
acimut=acimut+servo1[i];
altitud=altitud+servo2[i];

valor1=analogRead(sensor1);
valor2=analogRead(sensor2);
valor3=analogRead(sensor3);
valor4=analogRead(sensor4);

valor1=(float)lumenes(valor1);
valor2=(float)lumenes(valor2);
valor3=(float)lumenes(valor3);
valor4=(float)lumenes(valor4);

if(valor1> maximol){
    matriz[j][0]=valor1;
    maximol=valor1;
}
if (valor2> maximo2){
    matriz[j][1]=valor2;
    maximo2=valor2;
}
if (valor3> maximo3){
    matriz[j][2]=valor3;
    maximo3=valor3;
}
if (valor4> maximo4){
    matriz[j][3]=valor4;
    maximo4=valor4;
}
}
Servos(angulo);

aux++; //para hacer uno u otro
variable++; //para contar hasta 1 min
i++; //para el vector

if (variable>10){ //habra pasado un minuto
Serial.print(String(horas) + " hora ");
Serial.print(String(diferenciapos) + "dif posicion ");
Serial.println(String(maximol)+ " " + String(maximo2)+ " " + String(maximo3)+ " " +String(maximo4)+ " "
Serial.print(String(alt)+ " Serv altit " + String(movimiento) + " Serv acim ");
Serial.print(String(altitud)+ " Altitud ");
Serial.print(String(acimut)+ " Acimut ");
Serial.println(String(fin)+ " " );

minu=minu+1;
j++;
fin=fin+1;
aux=0;
variable=1;
maximol=0;
maximo2=0;
maximo3=0;
maximo4=0;
i=0;

if (minu>=60){
    hor=hor+1;
    minu=0;
}
}

```

ANEXO D: Código “sombras”

```

float latitud=38.013809*(PI/180); //UBEDA
float longitud=-3.381430;
float declinacion=-1;
int movimiento, alt;
float altitud=0;
float acimut=0;
int minu=5;
int hor=9+1;
int a=1;
int numero;
float maximo=116500;
byte matriz[305][4]={0};
int j=1;
int fin=0;
int maximo1,maximo2,maximo3,maximo4=0;
byte aux=0;
int i=1;
byte variable=1;
unsigned long periodo=3000;
unsigned long Tiempoahora=0;

const int sensor1=A1;
const int sensor2=A2;
const int sensor3=A3;
const int sensor4=A4;

    if(aux>=1){
        if(variable==20){
            matriz[j][0]=valor1;
            matriz[j][1]=valor2;
            matriz[j][2]=valor3;
            matriz[j][3]=valor4;
        }
        if ((valor1 - valor4)>6){
            i++;
        }
        if((valor4 - valor1)>6){
            i--;
        }
        if((valor3 - valor2)>6){

            a--;
        }
        if((valor2 - valor3)>6){

            a++;
        }
    }
    acimut=acimut+a;
    altitud=altitud+i;

    Servos(angulo);

```

```
aux++; //para hacer uno u otro
variable++; //para contar hasta 1 min
i++; //para el vector

if (variable>10){ //habra pasado un minuto
Serial.print(String(horas) + " hora ");
Serial.print(String(diferenciapos) + "dif posicion ");
Serial.println(String(maximol)+ " " + String(maximo2)+ " " + String(maximo3)+ " "
Serial.print(String(alt) + " Serv altit " + String(movimiento) + " Serv acim ");
Serial.print(String(altitud)+ " Altitud ");
Serial.print(String(acimut)+ " Acimut ");
Serial.println(String(fin)+ " " );

minu=minu+1;
j++;
fin=fin+1;
aux=0;
variable=1;
maximol=0;
maximo2=0;
maximo3=0;
maximo4=0;
i=0;
```