

Universidad Nacional de la Patagonia
San Juan Bosco



INFORME DE PROYECTO FINAL

SEGUIDOR SOLAR Y DEL PUNTO DE MÁXIMA POTENCIA **PARA UN PANEL FOTOVOLTAICO**

AUTORES: BARRÍA JUAN GABRIEL
VASQUEZ NICOLÁS ANDRÉS

TUTORES ACADÉMICOS: MAYOSKY MIGUEL
AHRTZ GERARDO

JULIO 2021

Índice

1 Introducción	Pág 2
2 Base Teórica	Pág 3
2.1 Celdas Fotovoltaicas : principio de funcionamiento	Pág 3
2.2 Métodos de fabricación	Pág 4
2.2.1 Celdas de Silicio amorfo	Pág 4
2.2.2 Celdas de Silicio monocristalino	Pág 5
2.2.3 Celdas de Silicio policristalino	Pág 6
2.3 Paneles fotovoltaicos: Módulos y arreglos	Pág 6
2.4 Modelo eléctrico	Pág 7
2.5 Curvas características	Pág 9
2.5.1 Curvas I – V	Pág 9
2.5.2 Curvas P – V	Pág 9
2.5.3 Eficiencia	Pág 11
2.6 MPPT	Pág 13
2.7 Algoritmos MPPT	Pág 14
2.7.1 Método de tensión constante	Pág 14
2.7.2 Método de tensión de circuito abierto	Pág 15
2.7.3 Método de corriente de cortocircuito	Pág 16
2.7.4 Método de Perturbar y Observar (P & O)	Pág 17
2.7.5 Método de conductancia incremental	Pág 19
2.8 Convertidor de Potencia	Pág 21
2.9 Tipos de Seguidores solares	Pág 26
3 Diseño y desarrollo	Pág 28
3.1 Panel fotovoltaico	Pág 28
3.2 Algoritmo MPPT	Pág 29
3.3 Sensor de tensión y corriente	Pág 29
3.4 Microcontrolador	Pág 29
3.5 Carga	Pág 30
3.6 Sensor de temperatura	Pág 31
3.7 LDR	Pág 32
3.8 Servomotor	Pág 32
3.9 Seguidor Solar	Pág 34
3.9.1 Programación	Pág 34
3.10 Diseño convertidor Buck	Pág 39
3.10.1 Simulaciones	Pág 41
3.11 Implementación algoritmo MPPT	Pág 43
3.12 Implementación algoritmo MPPT: Processing	Pág 46
3.12.1 Arduino y Processing	Pág 47
4 Anexos	Pág 50
4.1 Diseño de la estructura y engranajes	Pág 50
4.1.1 Geometría de engranes	Pág 50
4.1.2 Par de fuerzas y potencia transmitida	Pág 52
4.1.3 Calculo de engranes	Pág 53

4.2 Programa Principal	Pág 56
5 Bibliografía	Pág 62
6 Conclusiones	Pág 63

1 Introducción

La energía solar es una de las fuentes de energía renovable que más ha llamado la atención en los últimos años. Tanto es así que, actualmente, muchas de nuestras necesidades energéticas se satisfacen a partir del aprovechamiento de este recurso.

Es sabido que las fuentes de energía convencionales como el carbón, el gas natural y el petróleo se encuentran por naturaleza limitadas, ya sea en cantidad en el presente y en existencia a futuro. En cambio, el sol se presenta como una fuente inagotable de amplia disponibilidad. La energía solar es limpia y libre de emisiones tóxicas, ya que no produce agentes contaminantes o sub-productos nocivos para la naturaleza o los seres humanos. Por lo tanto, se trata de una industria en potencial desarrollo, donde constantemente se buscan nuevas y mejores formas de aprovechar este tipo de energía, y, por, sobre todo, de hacerla más eficiente.

La conversión de energía solar en eléctrica tiene muchos campos de aplicación. Hay dos formas principales en las cuales se transforma la energía solar en eléctrica: mediante colectores solares que utilizan la energía solar térmica, y con paneles fotovoltaicos que funcionan con energía solar fotovoltaica.

Con respecto a la energía solar térmica, se utiliza por ejemplo para calentar agua doméstica, así como para la generación de corriente AC en turbinas de vapor: el calor extraído de los rayos solares es usado para producir vapor, el cual se conduce a presión hacia un alternador para generar electricidad.

Por otro lado, los paneles fotovoltaicos están formados por celdas de materiales semiconductores, principalmente silicio o arseniuro de galio. Estas celdas convierten la energía de la luz absorbida en corriente DC mediante el efecto fotoeléctrico.

En este proyecto trabajaremos con este tipo de paneles, para los cuales se plantean dos objetivos principales:

- Implementar un circuito seguidor del punto de máxima potencia (MPPT) de la curva P-V característica, mediante un algoritmo adecuado.
- Implementar un circuito seguidor de la luz, con una estructura sobre la cual debe ir montado el panel, mediante resistencias LDR y servomotores.

2 Base Teórica

2.1 Celdas Fotovoltaicas: principio de funcionamiento

Las celdas o células fotovoltaicas están conformadas por dos tipos de materiales semiconductores, uno tipo “p” (que posee más huecos que electrones), y otro tipo “n” (posee más electrones que huecos). En un semiconductor expuesto a la luz, un fotón de energía arranca un electrón, creando a su vez un hueco en el átomo excitado. Esto es lo que se conoce como efecto fotoeléctrico. Normalmente, el electrón encuentra rápidamente otro hueco para llenar, y la energía proporcionada por el fotón se disipa en forma de calor.

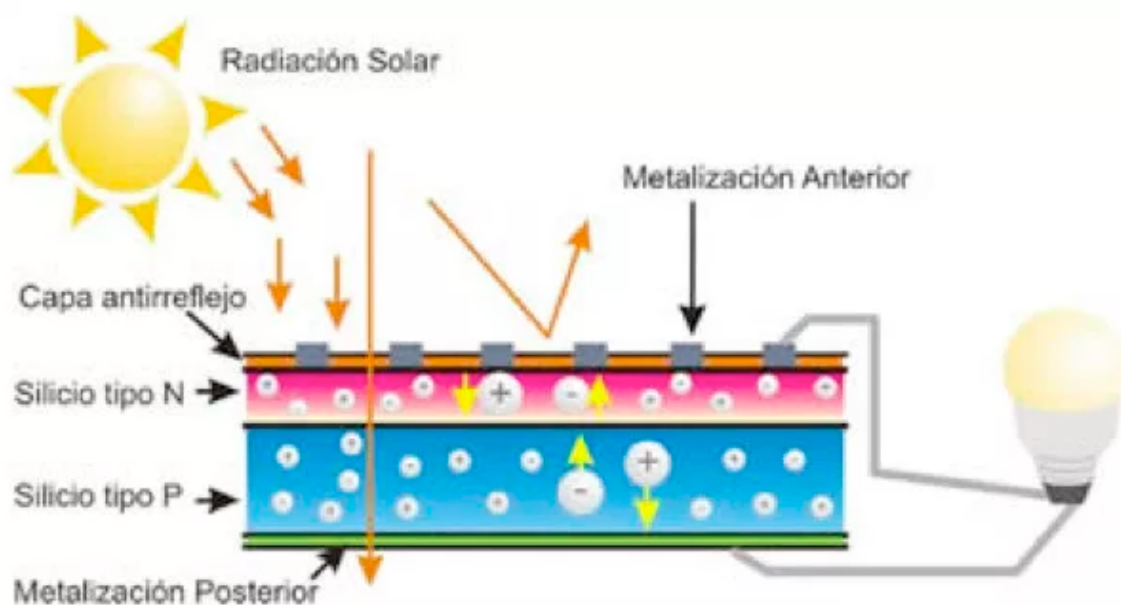


Fig. 2.1: Funcionamiento de una célula fotovoltaica

En cambio, el trabajo de una célula fotovoltaica es obligar a los electrones y huecos a avanzar hacia el lado opuesto del material, en lugar de solamente recombinarse en él. Así, se produce una diferencia de potencial y aparece tensión entre las dos partes del material, como ocurre en una pila. Para que esto suceda, se genera un campo eléctrico permanente, a través de la unión PN, entre las dos capas dopadas respectivamente. A esta zona se la llama zona de agotamiento.

La generación de corriente eléctrica ocurre dentro de la zona de agotamiento. Cuando un fotón choca contra la superficie y es absorbido por uno de los átomos en el material tipo N, desaloja un electrón, creando de esta manera un electrón libre y un hueco. El par electrón-hueco producido tiene la suficiente energía para salir de la zona de agotamiento. Si se conecta un cable desde el cátodo (material tipo N) al ánodo (material tipo P), los electrones fluirán a través del cable. Estos electrones son atraídos por la carga positiva del material tipo P y viajan a través de una carga externa, creando un flujo de corriente eléctrica. Los huecos creados a partir de los electrones libres son atraídos por la carga negativa del material tipo N y vuelven por el contacto eléctrico. A medida que los electrones ingresan en el material tipo P, se recombinan con los huecos, reestableciendo así la neutralidad eléctrica global.

2.2 Métodos de fabricación

Actualmente, el silicio es el material más comúnmente utilizado para la fabricación de celdas fotovoltaicas. Este material se obtiene por reducción de la sílice, que es el compuesto más abundante en la corteza terrestre, presente particularmente en la arena o el cuarzo.

El primer paso es la producción de silicio metalúrgico con un 98% de pureza, obtenido de piedras de cuarzo provenientes de un filón mineral. La pureza se logra mediante procedimientos químicos, empleando con frecuencia destilaciones de compuestos clorados de silicio. Para aplicaciones solares específicamente, son suficientes concentraciones de impurezas del orden de una parte por millón. Al material de esta concentración se le suele denominar silicio de grado solar.

Con el silicio fundido, se realiza un proceso de crecimiento cristalino que consiste en formar capas monomoleculares alrededor de un germen de cristalización. Nuevas moléculas se adhieren preferentemente en la cara donde se libera más energía durante el proceso. Las diferencias energéticas suelen ser pequeñas y pueden ser modificadas por la presencia de impurezas o cambiando las condiciones de cristalización. El germen de cristalización que provoca este fenómeno es extraído del silicio fundido, que se va solidificando en forma cristalina. Si el tiempo es suficiente, resulta un monocristal, y si es menor, un policristal. Este proceso se realiza a una temperatura superior a 1500 °C.

El silicio cristalino obtenido tiene forma de lingote. Estos lingotes son luego cortados en delgadas láminas de 200 micrómetros de espesor, llamadas obleas. Después del tratamiento de inyección del enriquecido con material dopante (Fósforo, Arsénico, Antimonio o Boro), se obtienen los semiconductores de silicio tipo N o tipo P.

Posteriormente, las obleas son metalizadas. Este proceso consiste en la incrustación de unas cintas de metal en su superficie y conectadas con contactos eléctricos. Estas cintas son las que transmiten la corriente eléctrica que generan las uniones PN a partir de la luz que reciben.

A partir del proceso de fabricación, se distinguen principalmente tres tipos de células fotovoltaicas: silicio amorfo, silicio monocristalino y silicio policristalino. Existe además un cuarto tipo, que es combinación de estos tres: células tándem.

2.2.1 Celdas de silicio amorfo

Durante su transformación, el silicio produce un gas que se proyecta sobre una lámina de vidrio, lo cual provoca que no se cristalice. Estas celdas pueden ser ligeras y flexibles según el tipo de material sobre el que se monte la capa de silicio amorfo. Además, poseen un color marrón o gris muy oscuro. Es el tipo de célula de las calculadoras y relojes solares.

Ventajas:

- Funcionan con luz difusa y baja (incluso en días nublados).
- Más económicas que las otras tecnologías.
- Integración sobre soporte rígido o flexible.

Desventajas:

- Rendimiento bajo a pleno sol, entre el 5% y 7%.

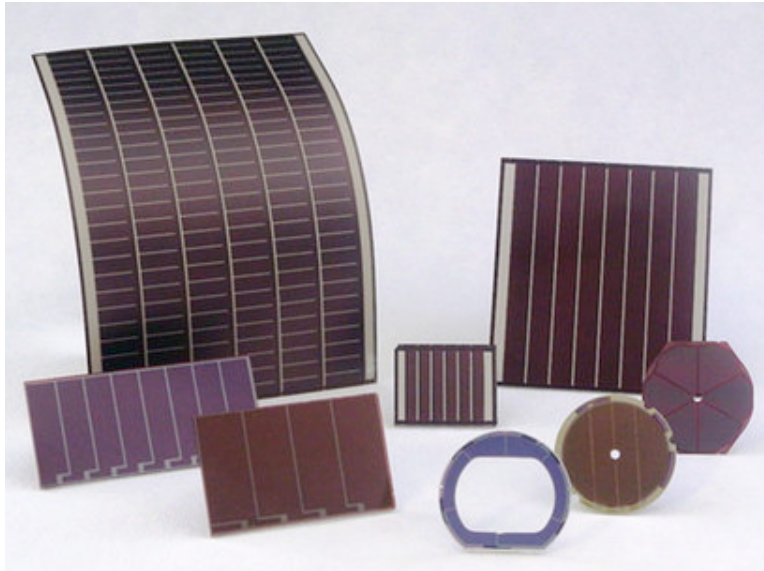


Fig. 2.2.1: Paneles hechos con células de silicio amorfo

- Rendimiento decreciente con el tiempo.

2.2.2 Celdas de silicio monocristalino

Al enfriarse, el silicio fundido se solidifica formando un único cristal de grandes dimensiones. Luego, se corta en delgadas capas de forma circular u octogonal, lo que da lugar a estas células. Como el corte se hace en una sola pieza, es bastante caro producirlas, pero son de las más eficientes que existen hoy en día. Poseen un color azul muy oscuro y uniforme.



Fig. 2.2.2: Panel hecho con células de silicio monocristalino

Ventajas:

- Buen rendimiento (15% al 25%).
- Buena relación potencia superficie, lo que ahorra espacio en caso de ser necesario.
- Número de fabricantes elevados.

Desventajas:

- Costo elevado.

2.2.3 Celdas de silicio policristalino

Durante el enfriamiento del silicio en un molde, se forma un conjunto de partículas cristalizadas de silicio. La célula obtenida es de aspecto azulado, pero no uniforme, ya que se distinguen varios colores creados por los diferentes cristales. Estas celdas deben montarse sobre marcos rígidos para protegerlas del deterioro. Su producción es menos costosa, pero resultan menos eficientes que las monocristalinas.

Ventajas:

- Células cuadradas que permiten mejor funcionamiento en módulo.
- Eficiencia de conversión óptima, pero un poco menor que las monocristalinas.
- Oblea más barata de producir que las células monocristalinas.

Desventajas:

- Rendimiento bajo en condiciones de iluminación pobre.

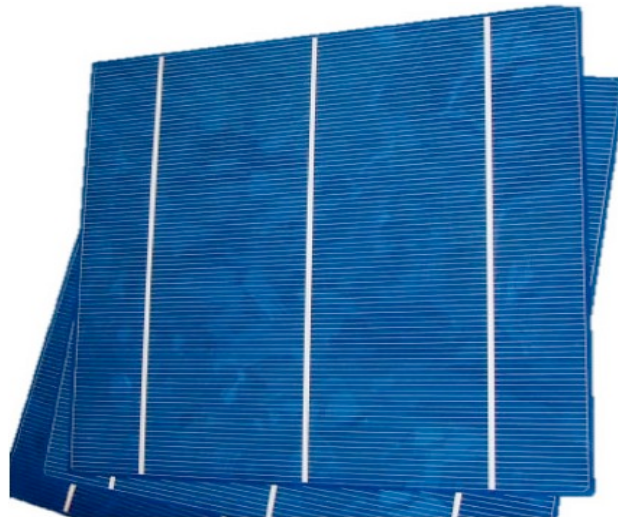


Fig. 2.2.3: Paneles hechos con células de silicio policristalino

2.2.4 Celdas tandem

Apilamiento monolítico de dos células individuales, por ejemplo, una capa delgada de silicio amorfo sobre silicio cristalino. Estas absorben en el espectro lumínico

al mismo tiempo, mejorando el rendimiento en comparación con las células individuales, sean amorfas, cristalinas o monocristalinas.

Ventajas:

- Gran sensibilidad en un amplio rango de longitudes de onda.
- Muy buen rendimiento.

Desventajas:

- El costo es alto debido a la superposición de dos células.

2.3 Paneles Fotovoltaicos: Módulos y Arreglos

Como vimos anteriormente, las unidades básicas de los sistemas fotovoltaicos son las celdas fotovoltaicas, las cuales producen una cantidad de potencia muy pequeña. Con la intención de aumentar la potencia obtenida, las celdas se interconectan entre ellas formando unidades mayores llamadas módulos. A su vez, los módulos pueden agruparse en arreglos fotovoltaicos, que producen aún más potencia. Conectando las celdas en serie, se aumenta la tensión obtenida. Además, si se conectan los módulos en paralelo, la corriente de salida alcanzará valores más altos.

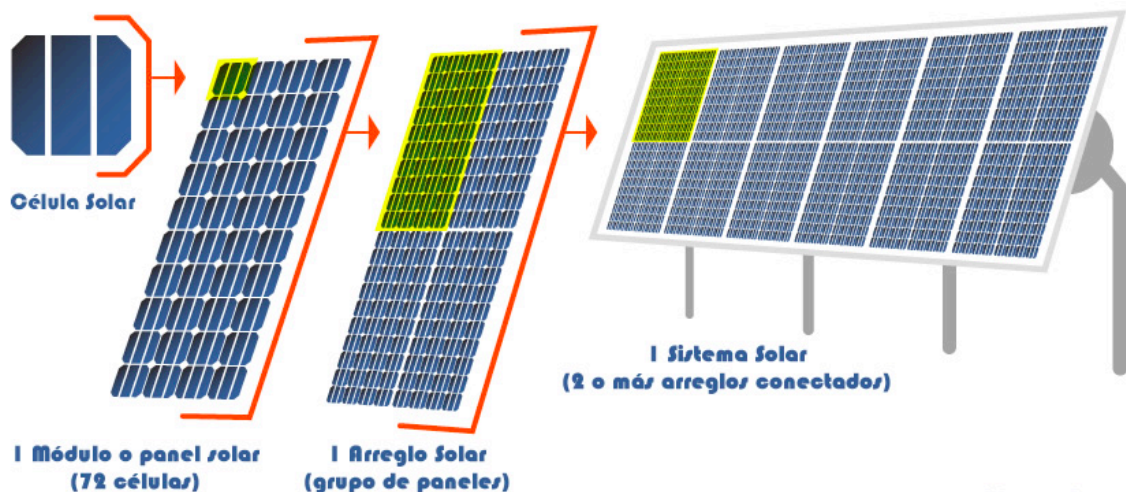


Fig. 2.3: Células, módulos y arreglos fotovoltaicos

2.4 Modelo Eléctrico

Las características de una celda fotovoltaica pueden ser explicadas con más detalle usando el circuito equivalente de la siguiente figura. El modelo consiste en una fuente de corriente, un diodo y dos resistencias. La fuente de corriente representa la corriente que es generada por la absorción de fotones en la celda, y su salida es constante en condiciones de temperatura e irradiancia constantes. La resistencia en serie R_s representa las pérdidas debido a contactos y conexiones. La resistencia en paralelo R_{sh} representa la corriente de fuga del diodo. Por último, se agrega una resistencia R variable conectada a la salida para simbolizar la carga.

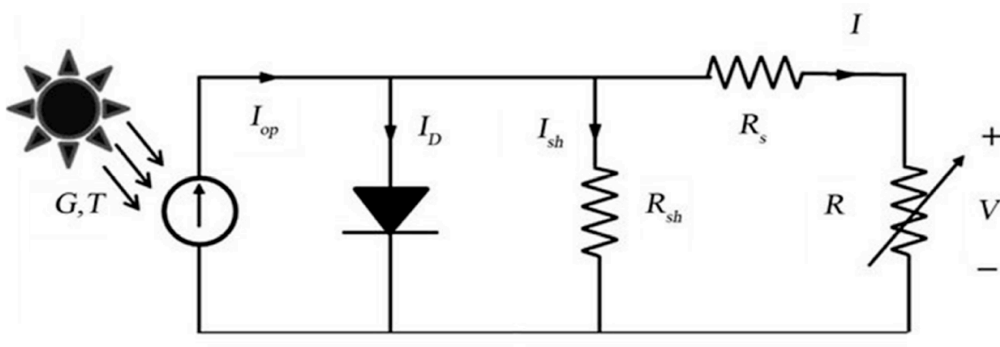


Fig. 2.4: Modelo eléctrico de una celda fotovoltaica

Observando el modelo se desprenden cuatro ecuaciones características:

- I_{op} : corriente fotoeléctrica.
- I_D : corriente del diodo.
- I_{sh} : corriente en la resistencia R_{sh} .
- I : corriente de salida total.

$$I_{op} = [I_{sc} + k_i(T - T_n)] \frac{G}{1000}$$

Para el diodo, primero se expresa la corriente de saturación inversa I_{rs} :

$$I_{rs} = \frac{I_{sc}}{e^{\frac{qV_{oc}}{nN_s kT}} - 1}$$

Entonces, la corriente en el diodo se expresa:

$$I_D = I_{rs} \left(\frac{T}{T_n} \right)^3 e^{\frac{qE_{go}(T_n^{-1} - T^{-1})}{nk}}$$

$$I_{sh} = \frac{V + IR_s}{R_{sh}}$$

De estas tres ecuaciones se obtiene la corriente I , que es la que en definitiva entrega la celda a la carga:

$$I = I_{op} - I_D - I_{sh}$$

$$I = I_{op} - I_{rs} \left[e^{\frac{q(V+IR_s)}{nkN_s T}} - 1 \right] - I_{sh}$$

Si hay varios módulos conectados en paralelo, esta última ecuación se modifica de la siguiente manera:

$$I = N_p I_{op} - N_p I_D \left[e^{\frac{q(V+IR_s)}{nkN_s T}} - 1 \right] - I_{sh}$$

	Descripción	Valor	Unidad
I_{sc}	Corriente de cortocircuito del módulo	Variable	A
k_i	Corriente de cortocircuito de la celda a una temperatura de 25°C y una irradiancia de 1000 W/m ²	3,2 (típico)	mA
T	Temperatura de operación	Variable	K
T_n	Temperatura nominal	298	K
G	Irradiancia solar	1000	W/m ²
q	Carga del electrón	$1,6 \cdot 10^{-19}$	C
V_{oc}	Tensión de circuito abierto del módulo	Variable	V
n	Factor de idealidad del diodo	1,3 (típico)	Adimensional
k	Constante de Boltzmann	$1,38 \cdot 10^{-23}$	J/K
E_{go}	Gap de energía del semiconductor	1,1 (típico)	eV
N_s	Cantidad de celdas conectadas en serie	Variable	Adimensional
N_p	Cantidad de módulos conectados en paralelo	Variable	Adimensional
R_s	Resistencia en serie	Variable	Ω
R_{sh}	Resistencia en paralelo	Variable	Ω

Tabla 2.4: Factores presentes en las ecuaciones del modelo eléctrico del panel fotovoltaico

2.5 Curvas Características

2.5.1 Curvas I-V

La curva característica I vs V se obtiene midiendo los distintos pares de valores Corriente – Tensión que surgen al variar la carga conectada al panel fotovoltaico. Estas mediciones se hacen con un nivel de luz fijo (irradiancia constante) y a una temperatura determinada. Se observan tres puntos característicos:

- Corriente de Cortocircuito (I_{sc}): es la corriente producida cuando se cortocircuitan los terminales de la celda o módulo, y la tensión en bornes es cero. Esto corresponde a la condición de carga cero.
- Tensión de Circuito Abierto (V_{oc}): es la tensión que aparece entre los terminales en condiciones de circuito abierto, cuando la corriente es cero. Esto corresponde a una carga infinita.
- Punto de Máxima Potencia (P_{Max}): también conocido como MPP (MaximumPower Point), es el punto donde la potencia obtenida es máxima. Cabe aclarar que la potencia del panel se calcula como el producto entre la corriente y la tensión que entrega el mismo.

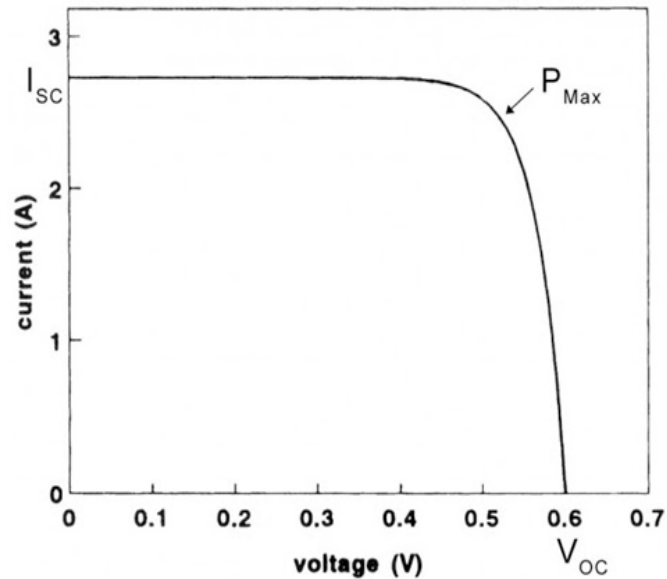


Fig. 2.5.1a: Curva Corriente vs Tensión

Como se mencionó anteriormente, los puntos de la curva se obtienen manteniendo temperatura e irradiancia constantes. En el caso de la figura, los valores corresponden a una irradiancia de 1000 W/m^2 y una temperatura de 25°C , que son las condiciones de prueba estándar (STC - Standard Test Conditions).

Si, por ejemplo, hay variaciones en la radiación solar, la curva sufre las modificaciones mostradas en la siguiente figura:

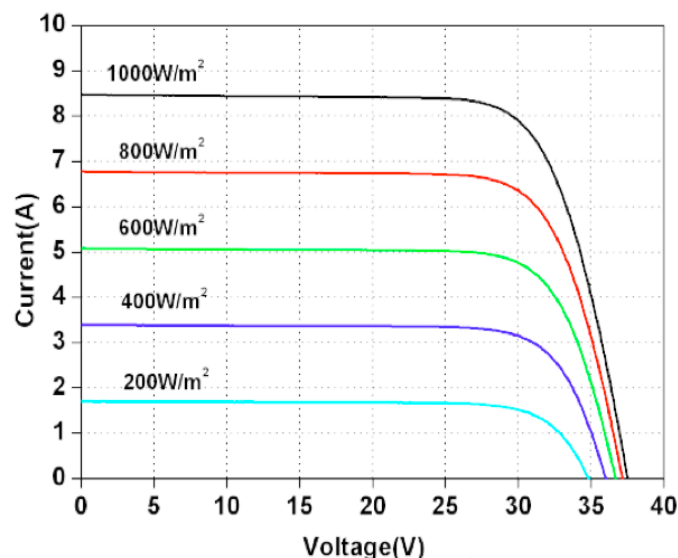


Fig. 2.5.1b: Curva I vs V para distintas irradiancias

Los cambios en la irradiancia provocan que, para una misma tensión, se obtengan corrientes más chicas, por lo tanto, la potencia también será menor (esto se mostrará más adelante). Para este caso, la temperatura se mantuvo fija en 25°C .

Por el contrario, si se producen cambios en la temperatura manteniendo constante la irradiancia, ocurre lo siguiente:

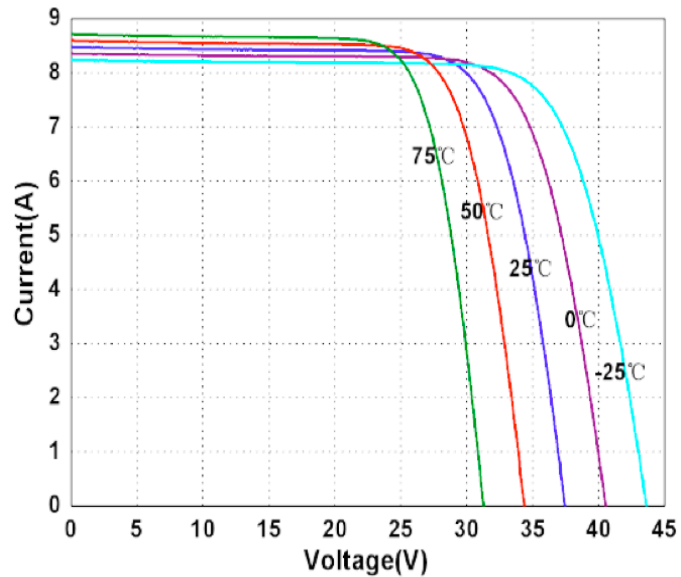


Fig. 2.5.1c: Curva I vs V para distintas temperaturas

2.5.2 Curvas P-V

Para determinar la forma de la curva P vs V, se multiplican punto a punto los valores de corriente y tensión de la curva I vs V, obteniendo lo siguiente:

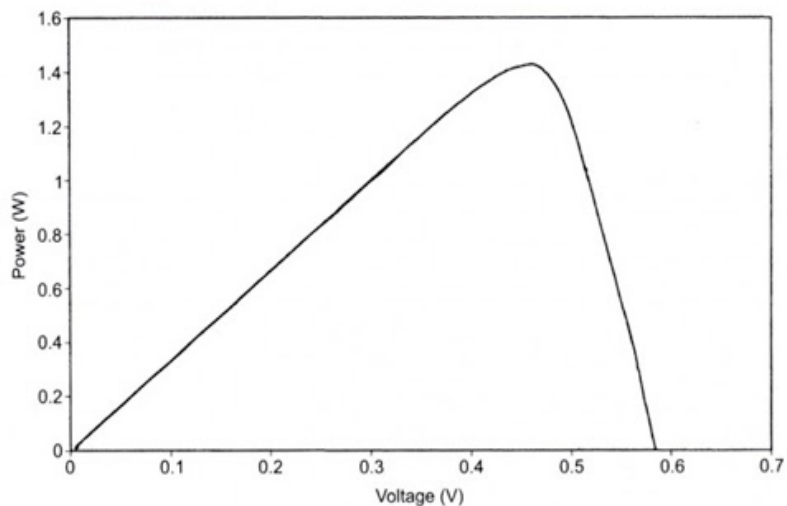


Fig. 2.5.2a: Curva Potencia vs Tensión

El pico de la curva indica el valor de potencia del punto de máxima potencia. Este valor es el producto de la tensión y la corriente para los cuales la potencia extraída del arreglo fotovoltaico es máxima. Por lo tanto, se les llama Tensión del Punto de Máxima Potencia (V_{MPP}) y Corriente del Punto de Máxima Potencia (I_{MPP}) respectivamente.

Como en el caso de la curva I vs V , al variar la irradiancia o la temperatura, la curva sufre las modificaciones mostradas a continuación:

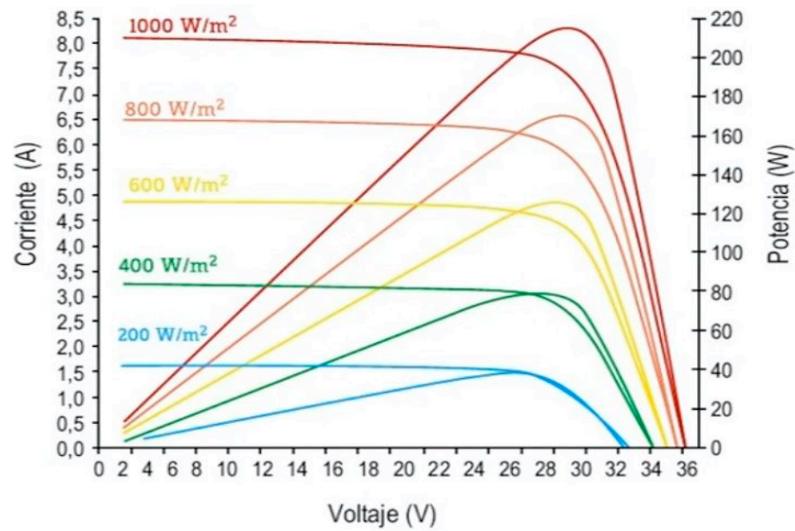


Fig. 2.5.2b: Curva P vs V para distintas irradiancias

En ambos casos, se observa que, si la irradiancia disminuye o si la temperatura aumenta, la potencia obtenida es menor. Por lo tanto, conviene siempre trabajar con la mayor irradiancia y la menor temperatura posibles, para maximizar la eficiencia del panel.

A continuación, se muestra la curva de Potencia y de Corriente contra Tensión, ambas en un mismo par de ejes, que es como se encuentran habitualmente.

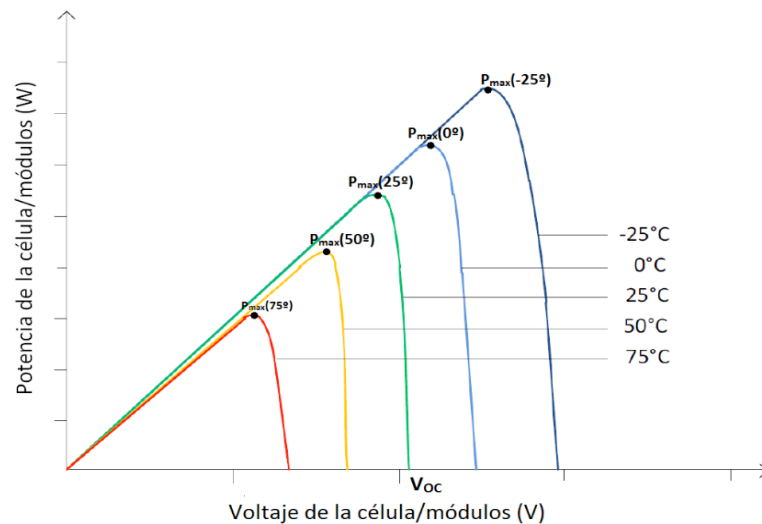


Fig. 2.5.2c: Curva P vs V para distintas temperaturas

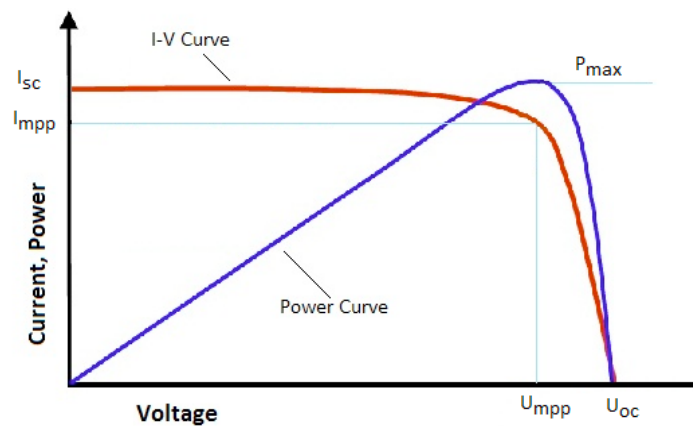


Fig. 2.5.2d: Curvas I y P vs V en un mismo par de ejes

2.5.3 Eficiencia

La eficiencia de una célula solar (η) es el porcentaje de la luz solar absorbida que se convierte en potencia eléctrica, cuando la célula está conectada a un circuito eléctrico. Este parámetro es el cociente entre el MPP, y el producto de la irradiancia con el área de la celda solar, bajo condiciones de prueba estándar (STC).

$$\eta = \frac{MPP}{G \cdot A_c}$$

Las STC corresponden a determinadas condiciones bajo las cuales se realizan las pruebas para determinar, de manera objetiva, los parámetros de referencia de los módulos fotovoltaicos. Estas condiciones especifican una temperatura de 25°C, una irradiancia de 1000 W/m² de espectro estándar y una masa de aire espectral (AM – Air Mass) de 1,5. La AM hace referencia a la cantidad de aire que debe atravesar la radiación solar para llegar a la superficie terrestre.

La eficiencia también queda determinada por la reflectividad que presenta el panel, y el gap de energía del material con el que está construido.

A esta altura se presenta un problema importante. Aun pudiendo trabajar en condiciones óptimas, con una irradiancia de 1000 W/m² y una temperatura lo más baja posible, las celdas fotovoltaicas presentan una eficiencia que ronda el 20%, y que en el mejor de los casos no supera el 25%. Por ejemplo, las celdas monocristalinas de silicio que se comercializan actualmente tienen una eficiencia máxima del 24%, que es intrínseca al material. Por lo tanto, hay entre un 75% y 80% de energía desaprovechada.

Otras de las restricciones existentes para mejorar la eficiencia son el ángulo de los paneles con respecto a la luz incidente, y la posible aparición de obstáculos (por ejemplo, nubes) que tapen la luz. Al hacer todos los ajustes necesarios de posición y ángulo se logra aumentar la eficiencia, pero no mantenerla a lo largo del tiempo.

2.6 MPPT

Para maximizar y mantener la potencia entregada por los paneles fotovoltaicos, se emplean algoritmos de Seguimiento del Punto de Máxima Potencia (MPPT – MaximumPower Point Tracking), los cuales se encargan de “rastrear” el MPP y mantenerlo frente a distintas condiciones de operación. Algunos de estos algoritmos funcionan básicamente de la siguiente manera: se mide tensión y corriente que entrega el panel y se calcula la potencia. Luego de un tiempo, se hacen las mismas mediciones y se calcula nuevamente la potencia. Las dos potencias obtenidas son comparadas, y se realizan los ajustes necesarios para que, en todo momento, la potencia más reciente tienda a la potencia máxima posible.

Existe una gran variedad de algoritmos MPPT, los cuales pueden ser agrupados en tres grandes grupos: Métodos Iterativos, Métodos Analíticos y Otros Métodos (como por ejemplo de Lógica Difusa). Estos algoritmos varían en su complejidad implementación, velocidad de convergencia al punto de máxima potencia, sensibilidad, sensores que requieren, costo y efectividad.

2.7 Algoritmos MPPT

A continuación, se hará una breve descripción de los siguientes métodos MPPT:

- Tensión Constante
- Tensión de Circuito Abierto
- Corriente de Cortocircuito
- Perturbar y Observar
- Conductancia Incremental

2.7.1 Método de Tensión Constante

Este es un método bastante simple pero ineficiente. Utiliza simplemente una única tensión para representar V_{MPP} . En algunos casos este valor queda determinado por una resistencia externa conectada a un pin del circuito de control. Para las distintas irradiancias, el método capta aproximadamente el 80% de la potencia máxima disponible. El valor actual se calcula con el nivel promedio de irradiancia. Como el MPP de un módulo fotovoltaico no siempre cae entre un 70% y 80 % de V_{OC} (tensión de circuito abierto) la eficiencia del seguimiento es baja en este caso.

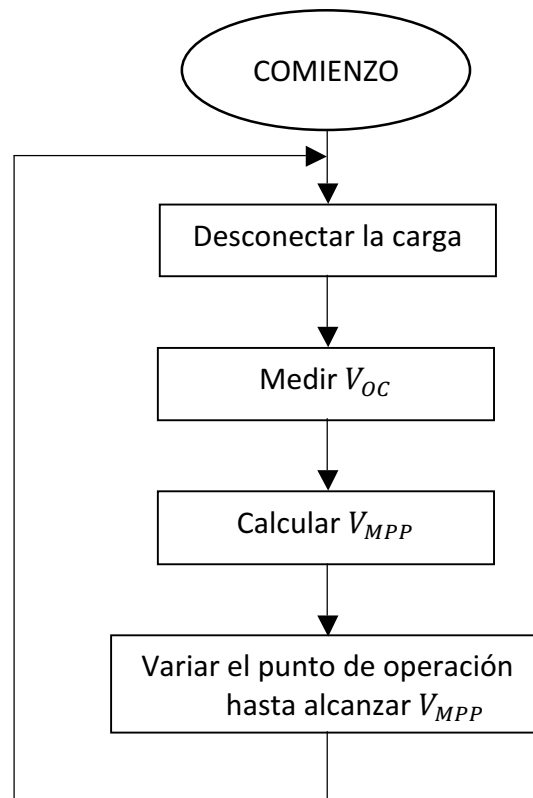


Fig. 2.8.1: Diagrama de flujo del método de Tensión Constante

2.7.2 Método de Tensión de Circuito Abierto

Es un método similar al anterior pero con mejoras, ya que usa V_{OC} para calcular V_{MPP} . Una vez que el sistema obtiene el valor de V_{OC} , V_{MPP} se calcula con la siguiente ecuación:

$$V_{MPP} = k * V_{OC}$$

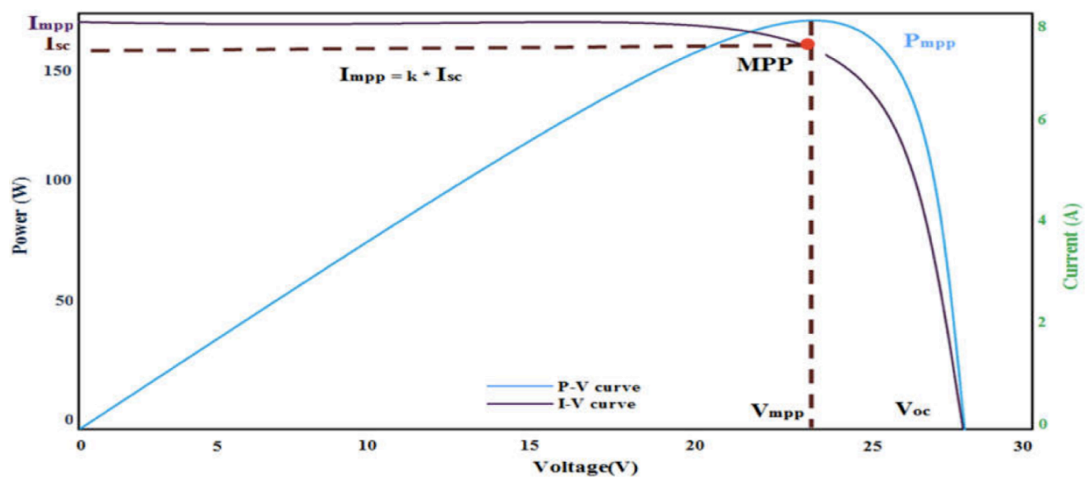


Fig. 2.8.2: Curvas del método de Tensión de Circuito Abierto

La constante k toma valores entre 0.7 y 0.8 y siempre es menor que 1. Comúnmente se adopta $k=0.76$. Esta constante depende de la función logarítmica de irradiancia, aumentando su valor a medida que la irradiancia aumenta.

Es necesario actualizar el valor de V_{OC} regularmente para compensar cualquier cambio en la temperatura. El tomar muestras de V_{OC} también puede ayudar a corregir pequeñas variaciones en la irradiancia. Además, se monitorea la corriente de entrada para saber cuándo se debe medir V_{OC} nuevamente.

Ventajas:

- El costo es relativamente bajo.
- Es un método bastante simple y fácil de implementar.

Desventajas:

- No es un método muy preciso y puede no operar exactamente en el MPP.
- El comportamiento del panel depende de la temperatura, por lo tanto, es necesario medir continuamente el valor de tensión de circuito abierto ante posibles variaciones en la temperatura.

2.7.3 Método de Corriente de Cortocircuito

También conocido como método de corriente constante. Este método usa un valor de I_{SC} (corriente de cortocircuito) para estimar I_{MPP} (corriente en el punto de máxima potencia). La I_{SC} es la corriente del panel con los terminales cortocircuitados. El valor de I_{MPP} se obtiene con la siguiente ecuación:

$$I_{MPP} = k * I_{SC}$$

Este método usa un pulso corto para generar la condición de cortocircuito. Durante el cortocircuito, la tensión caerá a cero, por lo tanto, el circuito convertidor de potencia debe ser alimentado desde otra fuente. El valor de k generalmente se encuentra entre 0.9 y 0.98, y siempre es menor a 1.

Ventajas:

- Es simple y el costo de implementación es bajo.
- Este método no requiere entradas.

Desventajas:

- No siempre es preciso.
- Los datos varían bajo diferentes condiciones meteorológicas y ubicaciones físicas.
- Tiene una baja eficiencia.

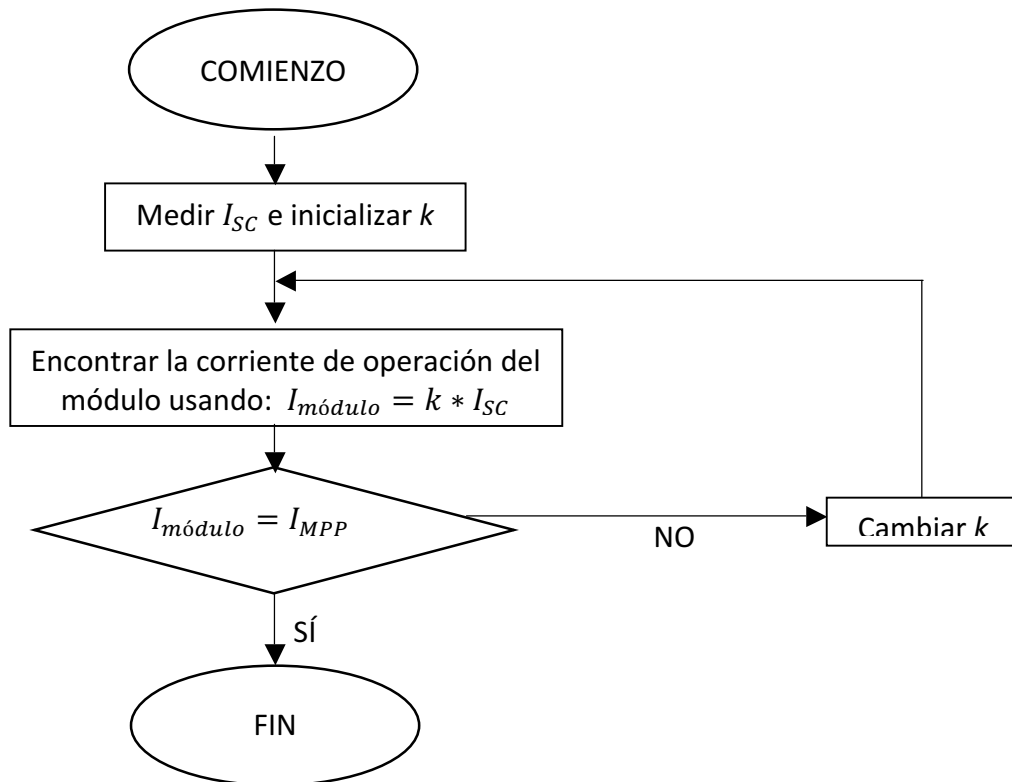


Fig. 2.8.3: Diagrama de flujo del método de Corriente de Cortocircuito

2.7.4 Método Perturbar y Observar (P&O)

El algoritmo P&O es ampliamente utilizado debido a su simplicidad de funcionamiento. En este método el controlador varía en una pequeña cantidad la tensión de salida y mide potencia. Si la potencia aumenta con respecto a la potencia anterior, entonces se sigue variando (perturbando) la tensión en la misma dirección hasta que la potencia ya no aumenta. Cuando la potencia empieza a disminuir, se perturba la tensión en la dirección contraria a la de un principio. Este proceso continúa hasta que se alcanza el MPP. En realidad, el resultado es una oscilación de la potencia alrededor del MPP.

El funcionamiento del método se puede explicar más claramente con las curvas características del panel. La potencia de salida del panel es función de la tensión del mismo, a una temperatura determinada y con irradiancia constante. En estas condiciones, se perturba la potencia (variando la tensión) y se observa el resultado. Si el cambio en la potencia es positivo, quiere decir que la potencia ahora es mayor que antes, entonces el punto de operación se ha movido más cerca del MPP. En este punto se debe examinar el cambio en la tensión. Si la diferencia de tensión entre la actual y la anterior es positiva, quiere decir que nos estamos aproximando al MPP por la izquierda. En este caso, se debe seguir aumentando la tensión (perturbando en sentido positivo). Si, por el contrario, el cambio en la tensión resultó negativo, esto significa que nos acercamos al MPP por la derecha. En este caso, se debe seguir disminuyendo la tensión (perturbando en sentido negativo).

Si desde un principio el cambio en la potencia es negativo, se debe invertir la perturbación en la tensión. Y, como último caso, si la potencia actual es igual a la anterior, se finaliza el algoritmo hasta una nueva iteración.

En la práctica, las perturbaciones se realizan básicamente variando el ciclo de trabajo (DC - DutyCycle) de un convertidor de potencia conectado a la salida del panel. Con el cambio del ciclo de trabajo, se modifica la corriente extraída del panel y la tensión en bornes del mismo, por lo tanto, la potencia extraída también se modifica en relación con el cambio en el ciclo del trabajo del convertidor.

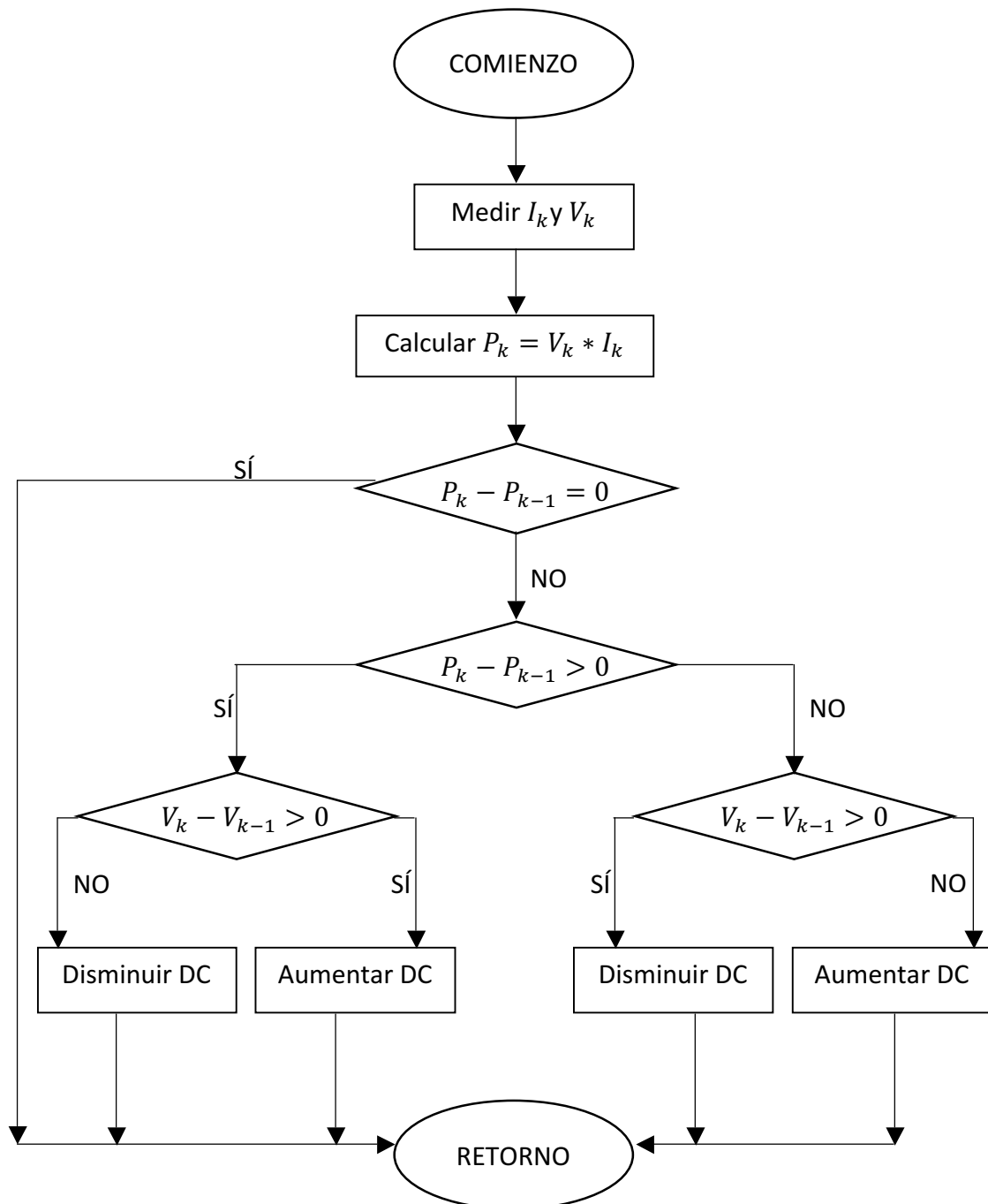


Fig. 2.8.4: Diagrama de flujo del método Perturbar y Observar

Ventajas:

- Algoritmo simple.
- Facilidad de implementación.
- Tiene un bajo costo de implementación comparado con los otros métodos.
- Es un método más preciso comparado con los anteriores.

Desventajas:

- No permite determinar realmente cuándo se ha alcanzado el MPP. Operando en estado estacionario, la salida oscila alrededor del MPP.
- El método es un poco lento para acercarse al MPP si la tensión está lejos del mismo.
- Si aparecen sombras sobre el panel, la curva de Potencia Vs Tensión presentará varios picos. Por lo tanto, el algoritmo no será capaz de distinguir el verdadero máximo, donde se encuentre el MPP.

2.7.5 Método de Conductancia Incremental

Este método surgió en el año 1993 con la intención de superar algunas dificultades que presentaba el algoritmo P&O. El MPP puede calcularse con la ayuda de la relación entre dI/dV y $-I/V$. El método se basa en el hecho de que en el punto de máxima potencia, la derivada de la potencia con respecto a la tensión es nula.

Matemáticamente, se tiene la siguiente ecuación:

$$\frac{dP}{dV} = \frac{d(V * I)}{dV} = I \frac{dV}{dV} + V \frac{dI}{dV} = I + V \frac{dI}{dV}$$

Como ya se mencionó, el MPP se alcanza cuando $dP/dV = 0$, entonces:

$$\frac{dI}{dV} = -\frac{I}{V}$$

$$\frac{dP}{dV} > 0 \text{ entonces } V_p < V_{mpp}$$

$$\frac{dP}{dV} = 0 \text{ entonces } V_p = V_{mpp}$$

$$\frac{dP}{dV} < 0 \text{ entonces } V_p > V_{mpp}$$

El procedimiento comienza midiendo los valores actuales de tensión y corriente del panel fotovoltaico. Luego, se calculan los cambios incrementales dI y dV , para lo cual se usan los valores actuales y los previos. La prueba se hace con las ecuaciones descritas anteriormente. Si la condición satisface la primera inecuación, se deduce que el punto de operación está a la izquierda del MPP, por lo tanto, debe ser movido a la derecha aumentando la tensión del panel. Análogamente, si la condición satisface la segunda inecuación, se deduce que el punto de operación está a la derecha del MPP, por lo tanto, debe moverse hacia la izquierda disminuyendo la tensión del panel.

Ventajas:

- El método es capaz de detectar satisfactoriamente cualquier cambio en la irradiancia y cambiar el valor del MPP ajustando el ciclo de trabajo.
- Tiene una buena eficiencia en el seguimiento.
- Reduce oscilaciones alrededor del MPP.
- Es posible reducir pérdidas de potencia y también costos en el sistema total.

Desventajas:

- Aumenta el tiempo de cómputo porque se reduce la frecuencia de muestreo de tensión y corriente. Esto se debe a que el método presenta una mayor complejidad comparado con el algoritmo P&O.

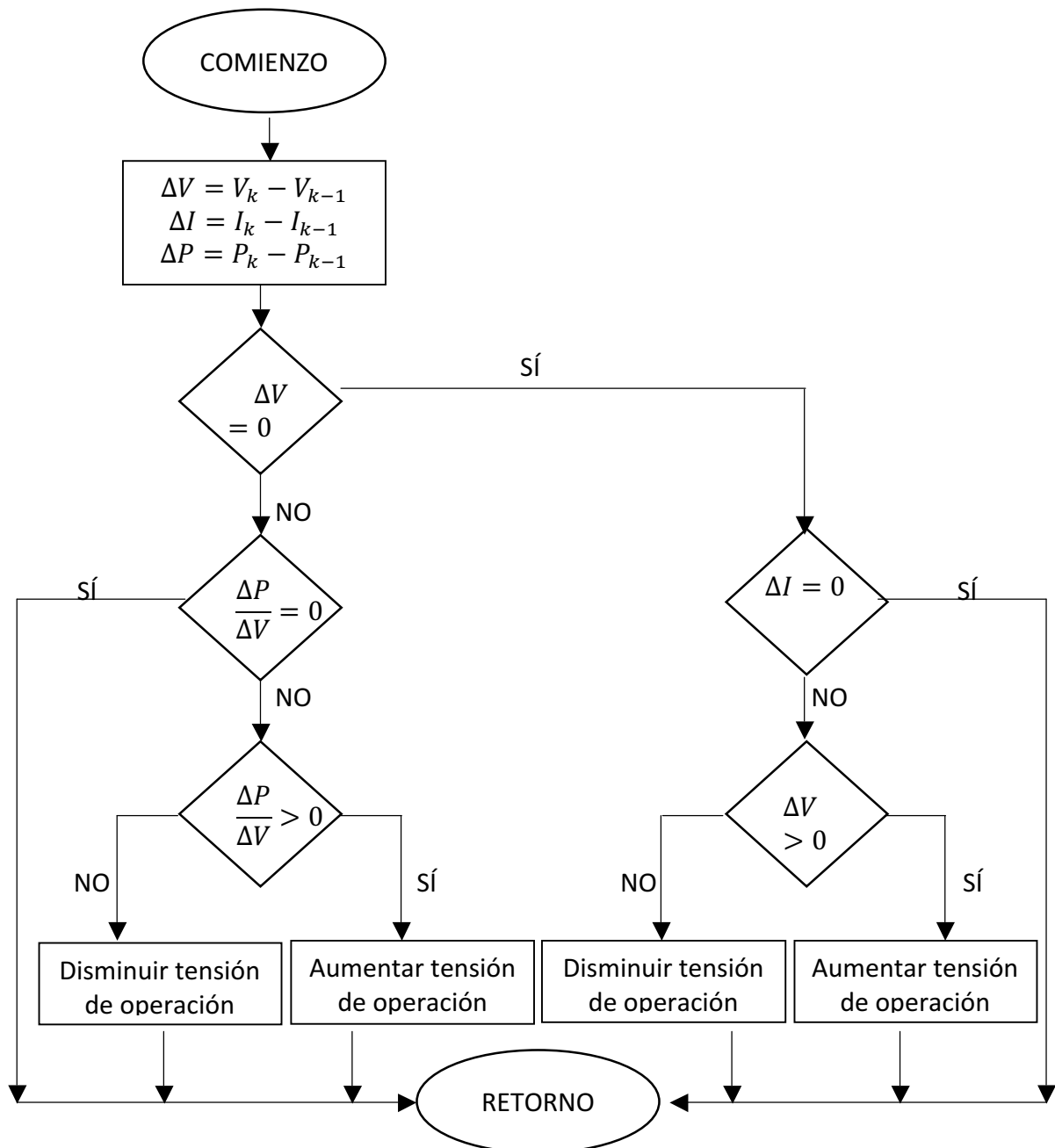


Fig. 2.8.5: Diagrama de flujo del método de Conductancia Incremental

2.8 Convertidor de Potencia

Como se mencionó en la sección anterior, las variaciones en tensión y corriente a la salida del panel se obtienen modificando el ciclo de trabajo de un convertidor de potencia.

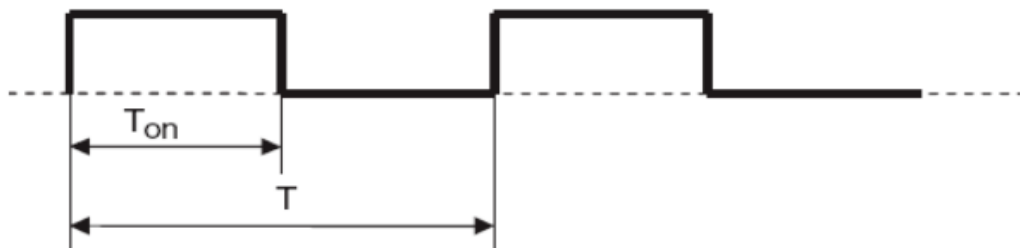


Fig. 2.9.1: Descripción gráfica del ciclo de trabajo

En este caso lo que se emplea es un “Regulador Conmutado a Frecuencia Propia” y lo que hace es convertir una determinada potencia de entrada proveniente de una fuente de corriente continua (la proveniente del panel solar, en nuestro caso) y convertirla a una potencia de corriente continua de magnitud variable (puede ser mayor, menor o igual) dependiendo de las condiciones de la carga.

Como se describió anteriormente, el regulador (que puede ser un transistor BJT, MOSFET o algún otro dispositivo que pueda trabajar como llave y de esta manera evitar una gran disipación de potencia siempre trabajando en sus zonas de corte y saturación) se podrá conmutar a una frecuencia controlada por el diseñador.

Los componentes básicos que componen estos tipos de reguladores son:

- Capacitor
- Inductancia
- Resistencia
- Transistor

En cualquier configuración de este tipo siempre estarán incluidos estos componentes variando la distribución de los mismos.

En nuestro caso se analizará el Convertidor Directo (ó Serie óBuck) el cuál provee de energía a la carga durante un período de tiempo (llave cerrada) y durante el otro período (llave abierta) la energía la provee la inductancia que almacenó energía. La tensión en la carga siempre será menor que la tensión de entrada y a lo sumo, idealmente, igual que la existente en la entrada.

La elección de este convertidor se basó teniendo en cuenta una posible aplicación a futuro de este proyecto, como cargador de baterías, para lo cual sería necesario reducir la tensión generada por los paneles fotovoltaicos comerciales a, por ejemplo, 6V o 12V.

Se define al ciclo de trabajo (δ) como la relación entre el tiempo que la llave está cerrada (T_{on}) y el período total (T)

$$\delta = T_{on}/T$$

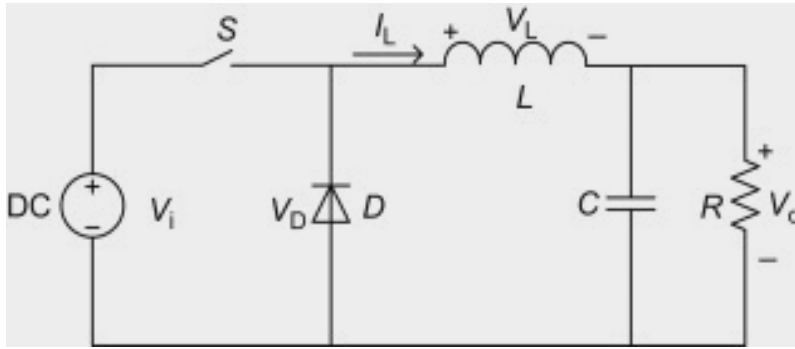


Fig. 2.9.2: Esquema circuital del Convertidor Buck

Donde el período T es la inversa de la frecuencia de conmutación del convertidor

$$T = 1/f$$

De esta manera, el ciclo de trabajo es un número que va de 0 a 1. Variando este parámetro, se pueden obtener distintos valores de tensión de salida (V_o) en el convertidor, para una tensión de entrada (V_i) constante.

Considerando el sistema periódico (estado estacionario), se debe cumplir que la tensión media en la inductancia L y la corriente media en el capacitor C deben ser iguales a :

$$\begin{aligned} \bar{V}_L &= 0 \text{ V; para que no varíe la corriente media } \bar{I}_L \\ \bar{I}_C &= 0 \text{ A; para que no varíe la tensión media } \bar{V}_C \end{aligned}$$

Además, considerando componentes ideales, la potencia disipada es cero por lo que:

$$P_i = \bar{V}_i * \bar{I}_i = P_o = \bar{V}_o * \bar{I}_o$$

Toda la potencia de entrada es entregada a la carga (idealmente).
La corriente instantánea en la inductancia será:

$$i_L = \frac{1}{L} \int V_L * dt$$

Y la tensión instantánea en el capacitor será:

$$v_C = \frac{1}{C} \int I_C * dt$$

Teniendo en cuenta lo anterior, ahora se analizará el circuito a implementar prestando atención a lo que serían las gráficas correspondientes a los distintos instantes de conmutación.

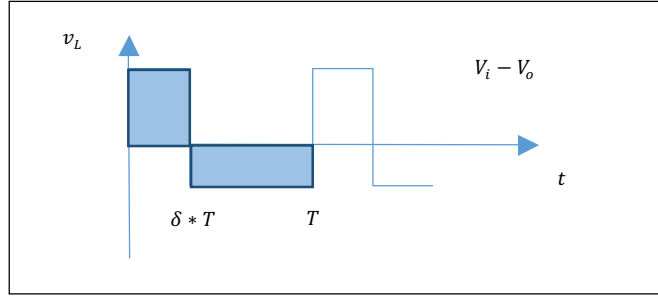


Fig.2.9.3 Tensión en la bobina

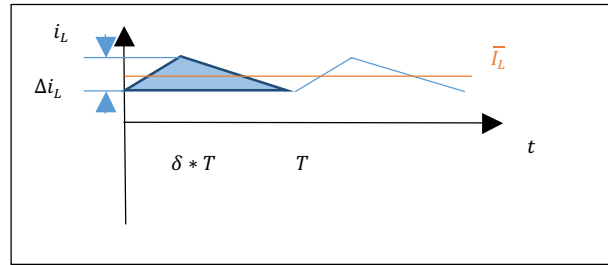


Fig.2.9.4 Corriente en la bobina

La primera gráfica se corresponde con la tensión instantánea y la segunda con la corriente instantánea, ambas en la inductancia.

Considerando el estado estacionario, se tiene que en el primer instante ($0 \leq t \leq \delta * T$) cuando la llave está cerrada el diodo se polariza en inversa y bajo la suposición de que el capacitor es lo suficientemente grande para que $V_o \approx \text{constante}$, la inductancia tendrá entre sus terminales una tensión de

$$i_L(t) = \frac{1}{L} * \int (V_i - V_o) * dt = \frac{V_i - V_o}{L} * t$$

Por lo tanto la tensión entre terminales de la inductancia será $V_i - V_o$ durante todo este período de tiempo.

En el instante en que se abre la llave ($\delta * T \leq t \leq T$), la inductancia genera una f.c.e.m oponiéndose al cambio de la corriente que por ella circula y en consecuencia el diodo se polariza en directa generando un camino por donde pueda circular la corriente en la bobina.

Ahora la inductancia quedará conectada en paralelo con la carga por lo que la tensión entre sus terminales será $-V_o$ y siguiendo el mismo análisis anterior se tiene que la corriente ahora será

$$i_L(t) = \frac{1}{L} * \int -V_o * dt = \frac{-V_o}{L} * t$$

Entonces sabiendo que $\bar{V}_L = 0 V$ se puede considerar que las dos áreas rectangulares que se tienen en la gráfica de v_L serán iguales por lo tanto

$$(V_i - V_o) * \delta T = -V_o * (T - \delta T)$$

$$\delta = \frac{V_o}{V_i} \quad \text{ó} \quad V_o = \delta * V_i$$

Esta es la ecuación de salida la cual deja en evidencia la naturaleza reductora de este tipo de convertidor, ya que, como se mencionó anteriormente, δ es un número que varía entre 0 y 1. Por lo tanto, V_o puede tomar valores entre 0 V y V_i como máximo.

Anteriormente se hizo la suposición de que $V_o \approx \text{constante}$ ya que en realidad existirá un ripple ΔV_o por lo que existirá una pequeña variación v_o cuyo valor medio $\bar{V}_o$ es el que en realidad medimos en la carga.

La suposición de que C es elevado es para poder considerar que la constante de tiempo $C * R \gg T$ y que la corriente media $\bar{I}_L$ pasa solo por R mientras que el ripple de corriente i_L pasa por el capacitor C . Esto se puede entender sabiendo que la corriente en la inductancia estará compuesta por un valor medio y un valor que varía con el tiempo o ripple

$$i_L(t) = \bar{I}_L + i_L$$

De esta manera se cumple que $\bar{I}_C = 0 \text{ A}$ y la cantidad de carga que ingresa al capacitor, ΔQ , es la misma que sale del mismo por lo tanto se puede hacer un análisis similar para el que se hizo con la tensión en la inductancia e igualar áreas ya que la corriente media es cero. Para ello detallamos la siguiente gráfica

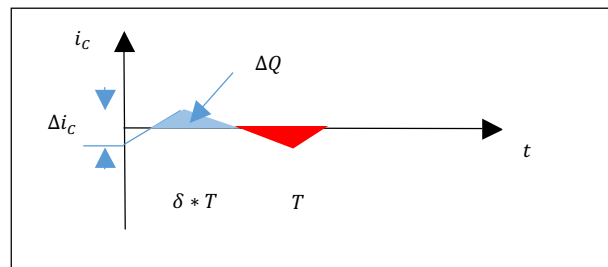


Fig.2.9.5 Corriente en el capacitor

Como se dijo anteriormente $\bar{I}_C = 0 \text{ A}$ por lo que se puede observar que la gráfica de corriente para el capacitor es la misma que la de la inductancia solo que desplazada hacia abajo sobre el eje de las ordenadas ya que el valor medio de corriente es cero.

Igualando el área positiva (celeste) con el área negativa (roja)

$$Q_C = C * V_o$$

$$\Delta V_o = \frac{\Delta Q_C}{C} = \text{Ripple de tensión en la carga}$$

Tomando el área positiva (también se podría haber tomado el área negativa), pero teniendo en cuenta la gráfica de corriente para la inductancia se tiene que en ese instante la corriente será

$$\Delta i_L = \delta * T * \frac{V_i - V_o}{L} * \frac{1}{2}$$

Esta será la altura del triángulo de área positiva y se divide por dos ya que, aunque se analiza, en este caso, el ripple de corriente en la inductancia hay que tener en cuenta que indirectamente se trata de la corriente en el capacitor que tiene valor medio cero. Además, hay que tener en cuenta la base del triángulo que en este caso será

$$b = \frac{T}{2}$$

Por lo que el área del triángulo quedaría

$$\Delta Q = \frac{b * \Delta i_L}{2}$$

Por lo tanto, el ripple de corriente quedaría igual a

$$\Delta V_o = \frac{V_i * (1 - \delta)}{8 * C * L * f^2}$$

Entonces para hacer la más chico posible al ripple lo que se puede hacer es incrementar C , L o f pero en la práctica C y L tomarían valores muy grandes por lo que lo mejor es variar la frecuencia de switcheo f porque además está elevada al cuadrado y solo la limitaría la frecuencia de conmutación del elemento regulador.

Para comprender la gráfica de la tensión en el capacitor hay que recordar que su corriente es una rampa creciente y decreciente ($i_C = m * t$; donde m sería la pendiente) por lo que al integral de esta rampa sería una cuadrática, la cual se observa en el siguiente gráfico

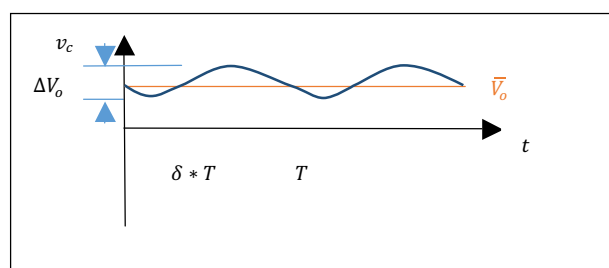


Fig.2.9.6 Tensión en el capacitor y la carga

Este análisis está hecho considerando un valor de corriente de salida I_o igual a la corriente media en la bobina $\bar{I}_L$, tal que la corriente instantánea de la bobina nunca llega a cero. A esta situación se le denomina **Condición de Conducción Continua**.

Si, por el contrario, la I_o es tal que la corriente instantánea de la inductancia se hace cero por un determinado período de tiempo, el convertidor pasa a funcionar en **Condición de Conducción Discontinua**, para la cual la ecuación de V_o descripta anteriormente deja de tener validez.

Así, surgen los denominados valores críticos, que son los valores de R o de L que, para una situación determinada de V_i , δ y f constantes, hacen que el circuito pase de Conducción Continua a Conducción Discontinua.

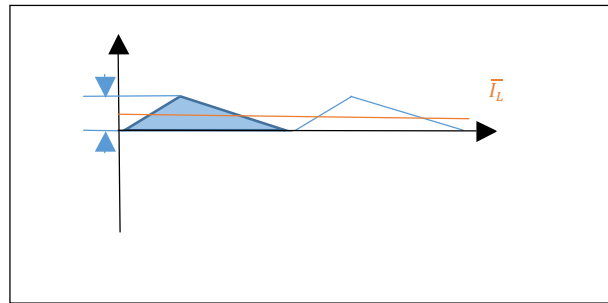


Fig.2.9.7 Corriente límite en la inductancia

Estudiando la gráfica anterior que representa el caso límite entre conducción continua y discontinua, se puede ver que ahora $\bar{I}_L = \frac{\Delta i_L}{2}$ además sigue teniendo validez la ecuación $V_o = V_i * \delta$ porque todavía no se está en la zona de discontinua.

$$\bar{I}_L = \delta * T * \frac{(V_i - V_o)}{2 * L} = \delta * T * \frac{(V_i - V_i * \delta)}{2 * L}$$

$$\bar{I}_L = \frac{V_o}{R} = \frac{V_i * \delta}{R}$$

Igualando ambas ecuaciones

$$\delta * V_i * \frac{(1 - \delta)}{2 * f * L} = \frac{V_i * \delta}{R}$$

Finalmente, las ecuaciones correspondientes para L y R serán:

$$L_{crítica} = \frac{R_{máx} * (1 - \delta)}{2 * f}$$

$$R_{crítica} = \frac{2 * f * L}{(1 - \delta)}$$

2.9 Tipos de Seguidores Solares

Dependiendo de su rango de movimiento se definen en básicamente dos tipos:

- Seguidores de un eje, los cuales solos tienen un grado de libertad por lo que su giro solo puede ser polar u horizontal.
- Seguidores de dos ejes, con dos grados de libertad permiten tener un seguimiento solar más preciso.

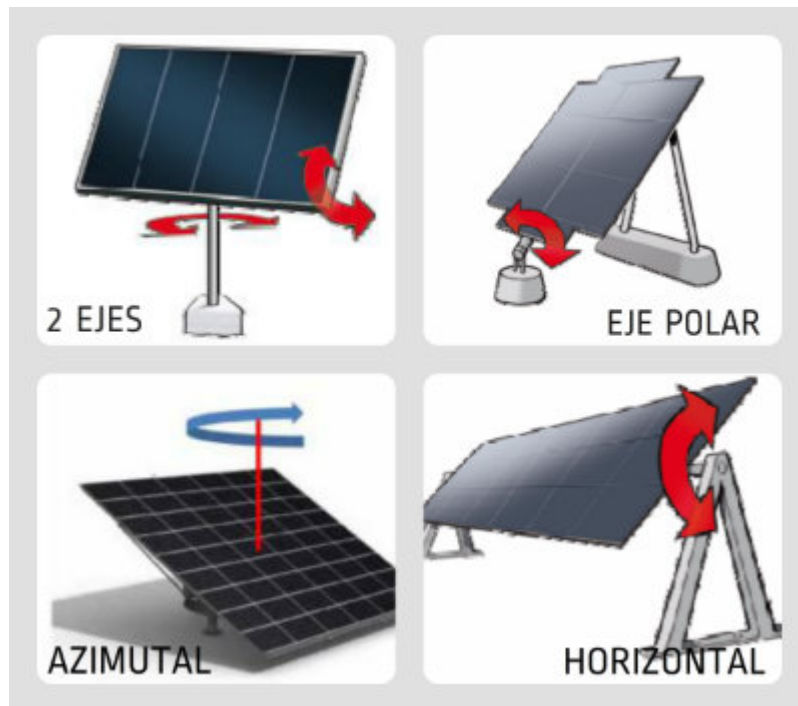


Fig.2.10.1

Dependiendo su algoritmo de seguimiento:

- Seguidores de punto luminoso, emplean un sensor (LDR por ejemplo) para indicar, en función de su nivel de tensión y posterior comparación, cuál es el punto de mayor iluminación.
- Seguidores de programación astronómica, se basan en algoritmos que predicen, en bases a determinadas ecuaciones y ajustes del operario, la posición del Sol en base a la época del año y considerando las condiciones climáticas del momento.

3 Diseño y desarrollo

3.1 Panel Fotovoltaico

Decidimos utilizar un panel de dimensiones físicas reducidas y poco peso, principalmente para facilitar la orientación del mismo hacia la luz. Optamos por el modelo CP-10M de la marca SolarLine, distribuido por la empresa Solar y Eólica SRL. Este panel está fabricado con celdas de silicio monocristalino, por lo cual presenta una alta eficiencia energética.

A continuación, se presentan las especificaciones del panel:

Potencia Máxima (P_m)	10 W (+/- 3%)
Corriente en P_m	0,57 A
Tensión en P_m	17,5 V
Corriente de Cortocircuito (I_{sc})	0,63 A
Tensión de Circuito Abierto (V_{oc})	21,6 V
Dimensiones Físicas	330 x 290 x 17 mm
Peso	1,5 kg

Tabla 3.1: Especificaciones del panel fotovoltaico a utilizar

Las especificaciones eléctricas se dan bajo condiciones de prueba estándar (STC), descriptas en la sección 2.6.

Este modelo presenta diodos superior e inferior incluidos de fábrica en el panel, los cuales evitan el indeseado efecto de punto caliente. Este efecto aparece cuando una sombra da sobre una parte del panel, provocando que las celdas pasen de generar energía a consumirla, comportándose como cargas eléctricas. Esto genera sobrecalentamientos en el módulo, y hasta la destrucción total de la celda.

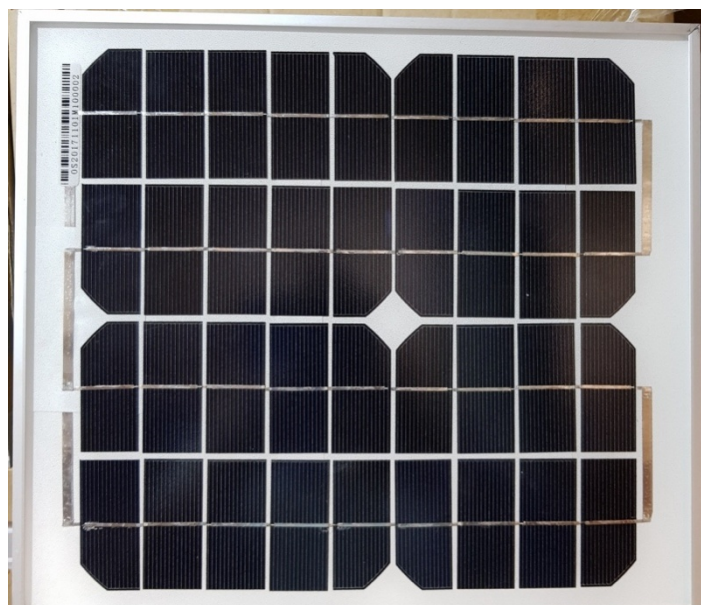


Fig. 3.1: Panel Fotovoltaico seleccionado

3.2 Algoritmo MPPT

Luego de analizar las ventajas y desventajas de los métodos descriptos, optamos por el Método Perturbar y Observar, por su simplicidad, facilidad y bajo costo de implementación. Aunque sabemos que la potencia oscilará alrededor del punto de máxima potencia, la precisión que ofrece el método es suficiente para nuestros objetivos.

3.3 Sensor de Tensión y Corriente

Para llevar a cabo el algoritmo MPPT, se requiere conocer los valores de tensión y corriente que entrega el panel. Obtendremos estos datos con un sensor dedicado a tal fin, en particular el INA219.

Este sensor tiene un amplificador de precisión que mide la caída a través de un shunt de 0.1 ohm, 1%. La máxima tensión diferencial admitida por el amplificador es de ± 320 mV, por lo tanto, puede medir corrientes de hasta ± 3.2 A. Combinado con el conversor analógico digital (ADC) de 12 bits, la resolución es de 0.8 mA. Además, permite medir tensiones entre 0 V y 26 V.

La lectura de corriente es directa, y un registro multiplicador interno calcula la potencia. La comunicación del INA219 con el exterior se realiza a través del protocolo I^2C . Si es necesario utilizar más de una placa a la vez, existe la posibilidad de asignarle una dirección única a cada una de ellas.

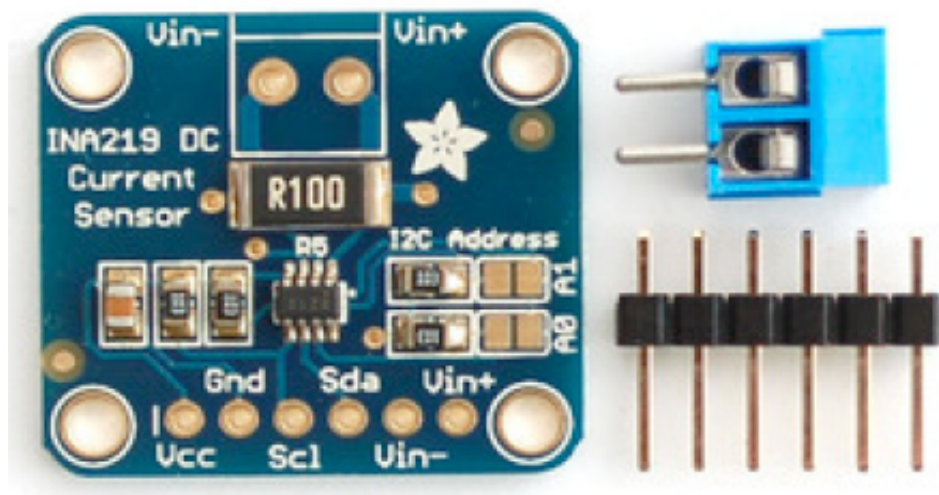


Fig. 3.3: Sensor INA219

3.4 Microcontrolador

Nuestro proyecto requiere un dispositivo que sea capaz de controlar el ciclo de trabajo del convertidor de potencia. El control del ciclo de trabajo se consigue mediante la variación del ancho de pulso de una señal cuadrada, lo que se conoce como Modulación por Ancho de Pulso (Pulse Width Modulation - PWM). De esta manera, la señal PWM se aplica al transistor del convertidor, que actúa como una llave electrónica.

Un microcontrolador es capaz de llevar esta tarea a cabo. Trabaja con la potencia calculada a partir de las muestras de tensión y corriente a la salida del panel. Ejecuta el algoritmo MPPT, y finalmente produce una modificación apropiada en la señal PWM de salida.

Particularmente, decidimos usar la placa Arduino Mega 2560, ya que disponíamos de ella previamente. Esta es una placa de desarrollo basada en el microcontrolador ATmega 2560. Provee 54 entradas/salidas digitales, de las cuales 15 pueden ser usadas como salidas PWM, 16 entradas analógicas, 4 UARTs, un cristal de 16 MHz, memoria Flash de 256 KB, jack para alimentación DC, conector ICSP, y botón de reseteo.

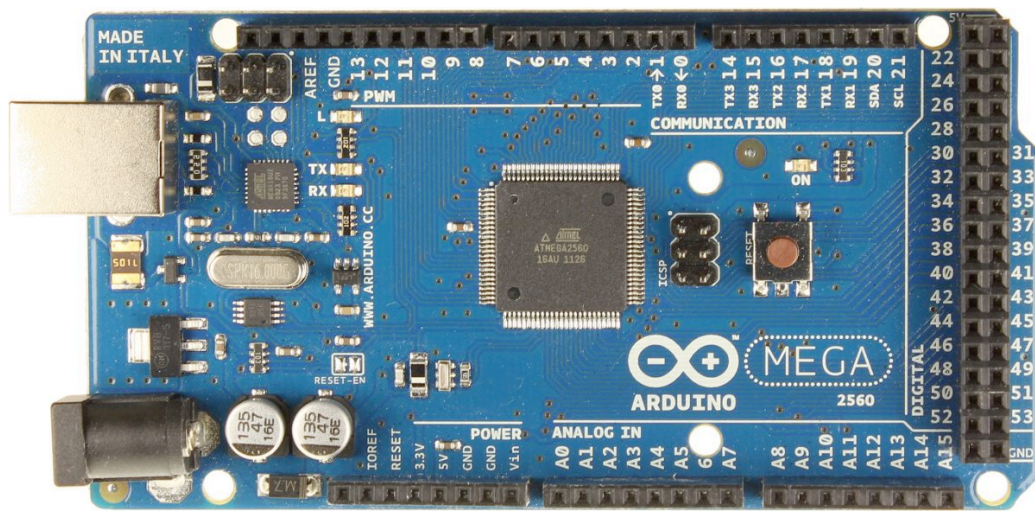


Fig. 3.5: Placa Arduino Mega 2560

La comunicación entre la placa y la computadora se establece través del puerto serie. Posee un convertidor USB – Serie, por lo que sólo se necesita conectar el dispositivo a la computadora mediante un cable USB como el que utilizan las impresoras.

Para la programación, se usa el entorno de desarrollo integrado (IntegratedDevelopmentEnviroment - IDE) de Arduino. La tarjeta viene preprogramada con un cargador de arranque (bootloader) que le permite cargar nuevo código en ella sin el uso de un programador de hardware externo.

El IDE presenta una función conocida como Monitor Serial, que permite visualizar los datos especificados en la pantalla. A partir de la versión 1.6.6, el IDE incluye el Serial Plotter, donde se pueden ver las variables gráficamente.

3.5 Carga

Como carga decidimos utilizar una resistencia de potencia. Al trabajar con un algoritmo MPPT, se garantiza que el panel siempre entregue la máxima potencia posible, independientemente del valor de carga que esté conectada.

Como se mencionó en la sección 3.6, una aplicación a futuro de este proyecto sería utilizar como carga una batería. La potencia del panel seleccionado es ideal para generar tensiones de salida de 12 V (utilizando el Buck), lo cual sirve para cargar baterías pequeñas y medianas, de amplio y variado uso como por ejemplo en sistemas de iluminación LED.

3.6 Sensor de Temperatura

Para la lectura de temperatura se emplea el sensor analógico LM35.

Este sensor, como cualquier otro, tiene unas características de funcionamiento que se detallan a continuación:

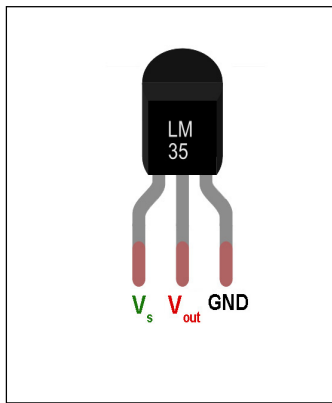


Fig. 3.7.1 Sensor

- V_s es la alimentación, normalmente a 5 V (pero su rango es de 4 a 30 V)
- La salida V_{out} está calibrada en grados centígrados ($^{\circ}C$) y los cambios de temperatura se dan de manera lineal de acuerdo a $10 \frac{mV}{^{\circ}C}$ por lo que por cada $1^{\circ}C$ tendremos un equivalente de 10 mV.
- Su rango va desde $-55^{\circ}C$ (-550 mV) y a $150^{\circ}C$ (1500 mV).

El Arduino leerá la temperatura de manera analógica y la transformará a su equivalente digital de 10 bits, por lo que, si alimento al sensor con 5 V tendré un rango de 0 a 1023 valores posibles de temperatura correspondiéndose el máximo valor, 1023, con la tensión de alimentación de 5 V.

Por lo tanto, teniendo en cuenta lo anterior y que $1^{\circ}C$ son 10 mV se podrá tener la siguiente ecuación:

$$T(^{\circ}C) = \text{valor leído por arduino}(0 \text{ a } 1023) * \frac{5V}{1024} * \frac{1}{10 \frac{mV}{^{\circ}C}}$$

$$T(^{\circ}C) = \text{valor leído por arduino}(0 \text{ a } 1023) * 5V * \frac{100^{\circ}C}{1023 V}$$

Pero de esta manera se podrán medir de temperaturas de hasta $500^{\circ}C$ a 5V, rangos que no podrá medir el sensor, por lo que, mediante una instrucción del programa, podemos cambiar la referencia para así poder mejorar la resolución en las medidas.

Para ello vamos a utilizar la función `analogReference` que nos permite establecer el valor de referencia para la entrada 1023. Hasta ahora hemos trabajado con 5V pero **si utilizamos INTERNAL podremos tener más resolución, 1.1V**. Esto equivale a poder medir temperaturas de hasta $110^{\circ}C$, dentro del rango de operación del LM35.

La ecuación sería la misma pero solo se modificaría el valor de 5V por 1.1V:

$$T(^{\circ}C) = \text{valor leído por arduino}(0 \text{ a } 1023) * 1.1V * \frac{100^{\circ}C}{1023 V}$$

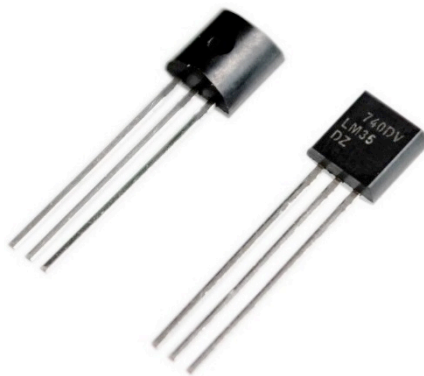


Fig. 3.7.2 Sensor LM35

3.7 LDR

Como se mencionó en la sección 2.6, el ángulo del panel fotovoltaico con respecto a la luz incidente es otro condicionante en la eficiencia energética. Para ello diseñamos una estructura móvil con dos ejes de movimiento, sobre la cual se montará el panel para orientarse hacia la luz.

Una forma sencilla de implementar un circuito seguidor de la luz es utilizando resistores que varían su valor de resistencia eléctrica dependiendo de la cantidad de luz que incide sobre su superficie. Estos son llamados fotorresistores o LDR (Light Dependent Resistor). La resistencia de un LDR es baja cuando hay luz incidiendo en él (en algunos casos puede ser tan baja como $50\ \Omega$), y muy alta cuando está a oscuras (del orden de varios $M\Omega$). Además, son capaces de reaccionar a una amplia gama de frecuencias, incluyendo infrarrojo, luz visible y ultravioleta.



Fig. 3.8: LDR (Light Dependent Resistor)

Se utilizarán cuatro LDRs en total, dos para cada plano de movimiento. Las señales de los fotorresistores se conectan directamente en los pines analógicos de la placa Arduino. Por programa se comparan las señales, y se decide cómo mover los servomotores para orientar el panel en la dirección deseada.

3.8 Servomotor

Un **servomotor** es un actuador rotativo o motor que permite un control preciso en términos de posición angular, aceleración y velocidad, capacidades que un motor normal no tiene. Utiliza un motor normal y lo combina con un sensor para la retroalimentación de posición.

Incluyen un motor de **corriente continua o alterna**, y son adecuados para su uso en un **sistema de control de bucle cerrado**. Se utilizan básicamente en la robótica industrial, en la fabricación con sistemas de automatización y en aplicaciones de mecanizado de control numérico (CNC) por ordenador.

El servomotor es un **servomecanismo** de bucle cerrado que utiliza la retroalimentación de posición para controlar su velocidad de rotación y posición. La señal de control es la entrada, ya sea analógica o digital, que representa el comando de posición final para el eje.

El **codificador o encoder** sirve como sensor, proporcionando retroalimentación de velocidad y posición. En la mayoría de los casos, sólo se informa de la posición. La posición final se informa al controlador y se compara con la entrada de posición inicial, y luego, si hay una discrepancia, se mueve el motor para llegar a la posición correcta.

Los servomotores más sencillos utilizan motores de corriente continua y detección de posiciones a través de un **potenciómetro** y también utilizan un control de gran potencia, lo que significa que el motor se mueve a la velocidad máxima hasta que se detiene en la posición designada. Estos son el tipo de motor que emplearemos en nuestro proyecto debido a su simplicidad para manejarlo y controlarlo.

Los servomotores se controlan enviando un pulso eléctrico de ancho variable, o modulación de ancho de pulso (PWM), a través del cable de control. Hay un pulso mínimo, un pulso máximo y una frecuencia de repetición. Por lo tanto, el servomotor espera ver un pulso cada 20 milisegundos (ms) y la longitud del pulso determinará hasta dónde gira el motor. Por ejemplo, un pulso de 1.5ms hará que el motor gire a la posición de 90° (esto último dependerá del servomotor, ya que no todos se desplazarán a un mismo ángulo para un ancho de pulso determinado, por lo que los desplazamientos reales hay que encontrarlos mediante la prueba de los mismos mientras se varía la PWM).

Pero básicamente el funcionamiento es el mismo para todos ya que en cada posición angular el servo la mantendrá para un ancho de pulso determinado que le envía, en nuestro caso, el Arduino y se desplazará una cierta cantidad de grados en función de cómo varíe ese ancho de pulso.

En nuestro caso se emplea el llamado servo posicional ya que solo gira desde 0° a 180° debidos a unos topes físicos que se le agregan a los engranes. Contrario a estos están los de rotación continua que no tienen ningún limitante respecto al giro.

Ambos motores pueden girar en ambos sentidos.

Para este proyecto, utilizamos el servomotor MG995, el cual es un servo económico, pero con alto torque y de alto rendimiento. Cuenta con un diseño robusto y dimensiones físicas pequeñas, lo cual es ventajoso para nuestra aplicación. Además, presenta engranes metálicos, lo que lo hace más resistente en comparación a otros servos que poseen engranes de plástico.



Fig. 3.9: Servomotor MG995

Especificaciones:

- Tensión de operación: 4.8 V a 7.2 V
- Velocidad de operación: 0.2 s/60° (a 4.8 V), 0.16 s/60° (a 6 V)
- Torque detenido: 8,5 kgf·cm (a 4.8 V), 10 kgf·cm (a 6 V)
- Dimensiones: largo 40.7 mm, ancho 19.7 mm, alto 42.9 mm (aprox.)
- Peso: 55 g

3.9 Solar Tracker o Seguidor Solar

En esta parte del proyecto lo que se hizo fue diseñar la estructura correspondiente del seguidor (engranes y soportes) para posteriormente poder agregar los servos y LDR.

Los LDR, como se detalló anteriormente, en función de la cantidad de luz que incida sobre los mismos variará su resistencia de manera no lineal generando así una correspondiente variación de tensión entre sus terminales.

Esta variación de tensión será captada y comparada mediante el Arduino, el cual enviará un pulso de determinada duración en una de sus salidas PWM en función de la lógica de comparación propuesta y hará que el servo correspondiente (azimut o vertical) lo transforme en un desplazamiento angular.

3.9.1 Programación

Para comprender un poco lo que sería la programación se empieza por detallar lo que sería el conexionado de los componentes que integran el circuito

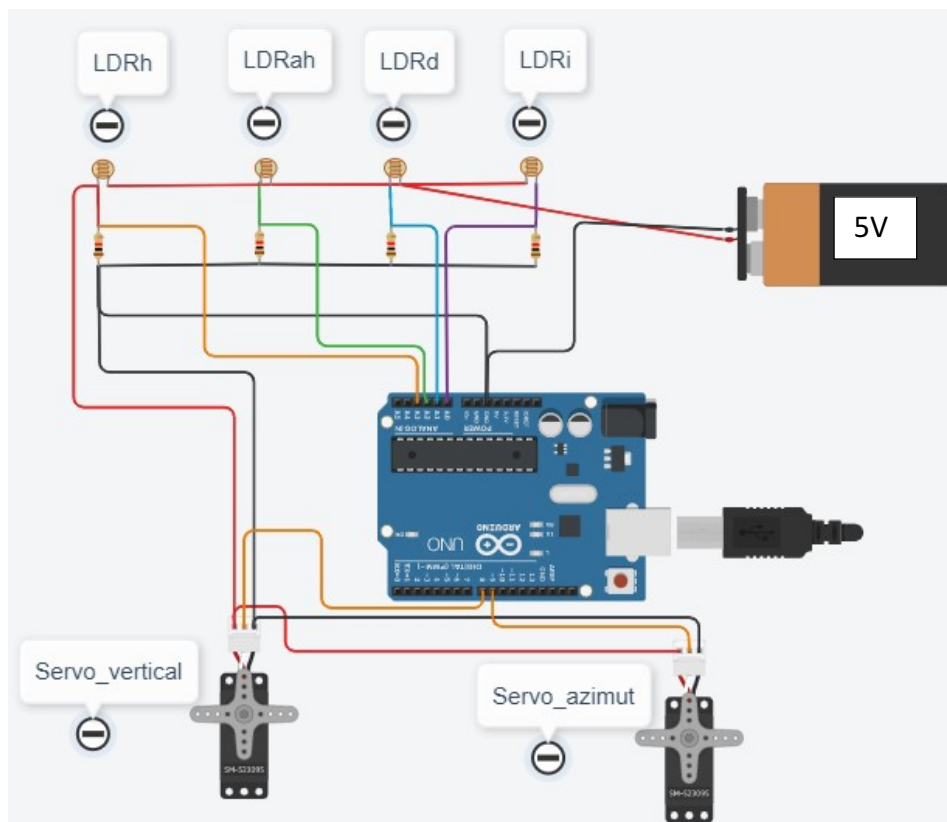
Los LDRh (giro horario) y LDRah(giro antihorario) son los que en función de su nivel de tensión controlan, mediante Arduino, el servo de giro vertical mientras que

LDRd (giro hacia la derecha) y LDRi (giro hacia la izquierda) hacen lo mismo pero sobre el servo azimuth.

La programación es la misma para ambos servos, por lo que solo se detallará la programación del servo vertical ya que para el servo azimuth solo cambian las entradas y los nombres de las variables en juego.

Para la programación lo que primero se hizo fue darle una posición inicial tanto al servo vertical como el servo azimuth ya que al conectar todo el circuito estos se posicionaban automáticamente en 90° y en consecuencia se producía una gran inercia en toda la estructura, lo que la volvía inestable. El ángulo inicial fue de 8° mediante la instrucción `Servo.write(8)` y mantenían esta posición hasta que se ejecutara otra instrucción.

Luego mediante un ciclo `for` se llevó tanto al servo de giro vertical como el azimuth a 60° y 45° respectivamente, de manera tal de poder tener una posición inicial para poder realizar esta parte del proyecto.



3.10.1 Esquema de conexionado circuital

A estos ángulos iniciales se los denominó como $\text{angmant1}=60^\circ$ y $\text{angmant2}=45^\circ$ ya que al entrar en función `void loop()` del programa se trabaja con la variable `ang1` y `ang2` que toman como valor inicial estos valores de manera tal de poder tener una referencia al momento de ejecutar el programa ya que otra manera de hacer esto es leer la posición actual de cada servo mediante la instrucción `Servo.read()` pero esta nos daba un valor erróneo de la posición verdadera del servo.

Hecho esto y ahora analizando solo al servo vertical, lo que se hizo fue leer los valores de tensión de los LDR en función de la luz que incide sobre los mismos,

entonces una vez que se tenían estos valores se procedió a hacer un delta de tensión que se denominó $dif1 = LDRh - LDRah$.

Para este $dif1$ se establecieron tres valores posibles de tensión, pero considerando que el Arduino tiene una resolución de 10 bits tendrá 1024 valores posibles desde 0 a 1023 para representar los 5V que alimenta a los LDR.

Sabiendo esto los tres casos posibles de $dif1$ serán:

- $-10 \leq dif1 \leq 10$, que es un límite que se estableció en función de distintas pruebas que se hicieron sobre los servos y es en este rango de valores donde los servos mantendrán su posición.
- $dif1 \geq 10$, indicará que $dif1$ es positivo y que la tensión de LDRh es mayor por lo que el ángulo $ang1$ se incrementa y el panel gira en sentido horario.
- $dif1 \leq -10$, indicará que $dif1$ es negativo y que la tensión de LDRah es mayor por lo que el ángulo $ang1$ se decrementa y el panel gira en sentido antihorario.

Esto sería las condiciones principales, pero hay que tener en cuenta que el servo solo puede girar desde 0° a 180° que son los límites ideales, ya que el servo al iniciar arranca en una posición que no son los 0° y que su giro máximo no son los 180° si no que son menos grados.

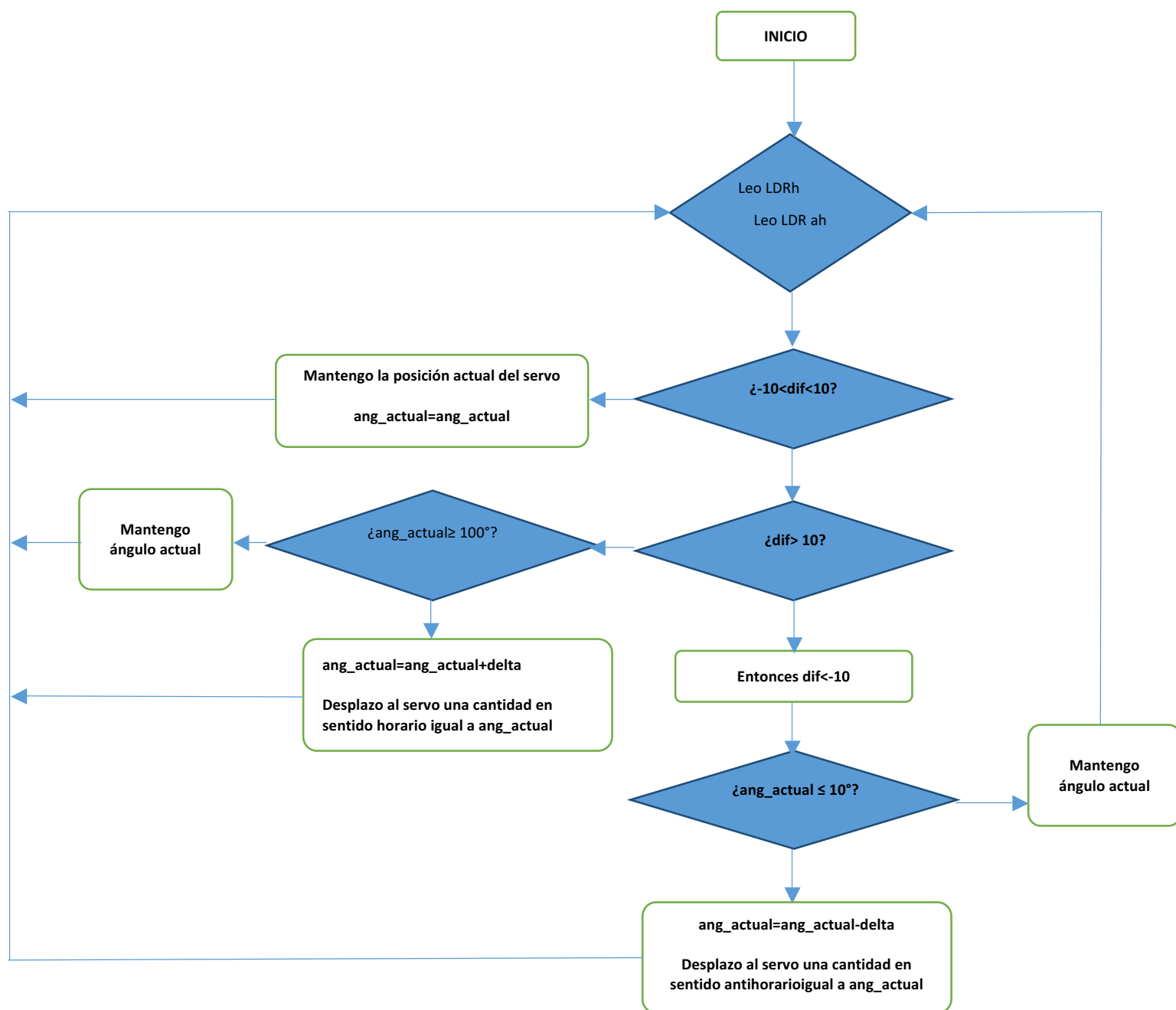
Además de no considerar estos límites extremos, al correr el programa y si los servos llegasen a estas posiciones límite, mantendrán esta posición y no responderán a las instrucciones que se les esté indicando además de producirse daños en sus engranes ya que lo que estaría sucediendo es que los servos estarán “intentando” girar más allá de sus valores extremos.

Considerando lo anterior se establecieron como valores extremos luego de varias pruebas

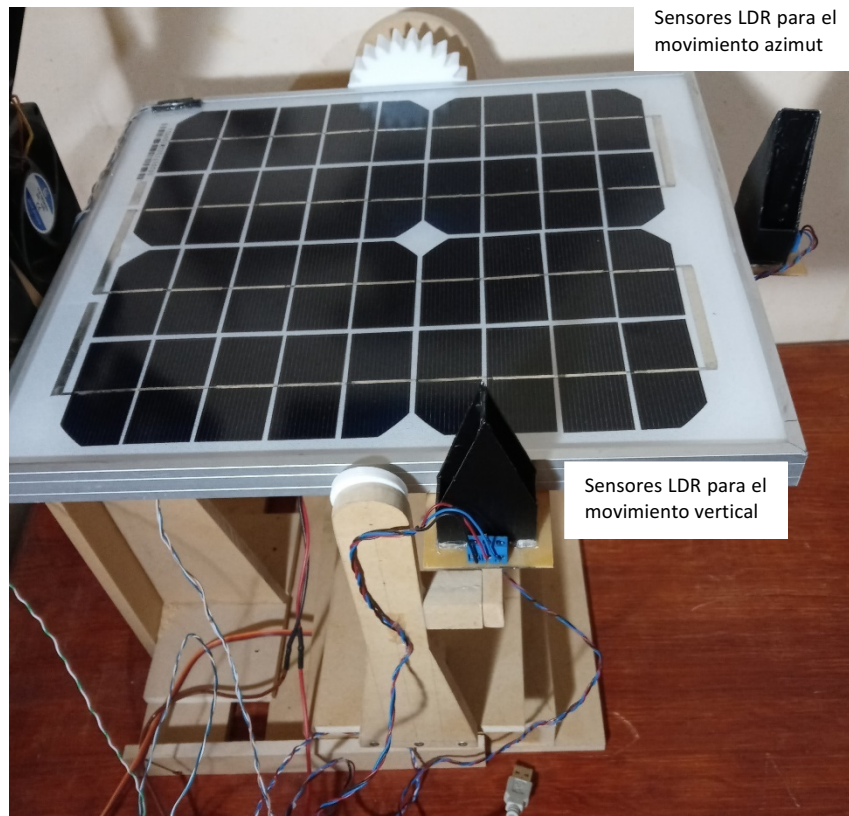
- Ángulo mínimo que se le permite llegar será de 10° .
- Ángulo máximo que se le permite llegar será de 100° .

Lo detallado anteriormente sería la manera en que se consideran los diferentes factores para poder llevar a cabo el proyecto, lo que se presenta a continuación sería el diagrama de flujo en el cual nos basamos para luego poder plasmarlo en la programación de Arduino.

Nuevamente, este diagrama de flujo es el correspondiente para el servo vertical y por las razones explicadas anteriormente no se detalla el correspondiente al servo azimuth ya que sería redundante.



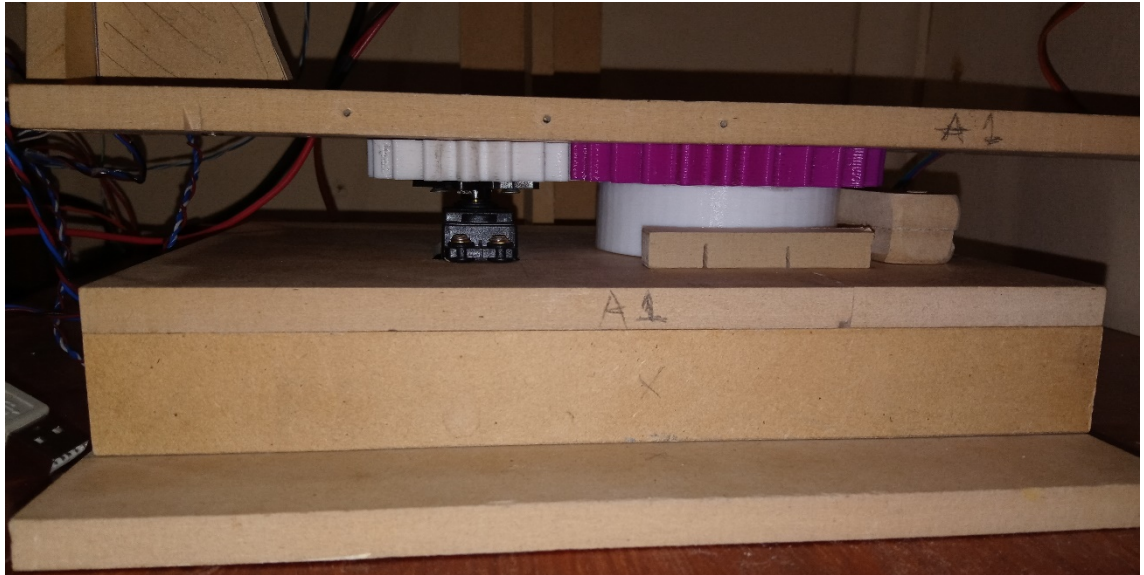
El programa que se hizo para Arduino se deja en la parte de anexos.



3.10.2 Esquema de conexionado LDR y estructura para el seguidor.



3.10.3 Servo que produce el desplazamiento vertical del panel.



3.10.4 Servo que produce el desplazamiento azimutal del panel.

3.10 Diseño Convertidor Buck

En primera instancia, el factor determinante para diseñar el convertidor fue el valor de inductancia de la bobina. Decidimos primeramente buscar entre las bobinas disponibles en el pañol de nuestro laboratorio, teniendo en cuenta que un valor útil de inductancia para una implementación práctica tenía que estar por encima de 100 μH . Para esto, utilizamos el analizador de impedancias del laboratorio y trabajamos con una frecuencia de 31.25 kHz. Este valor surge de la frecuencia de trabajo del Timer 2 de la Arduino Mega, el cual controla los pines 9 y 10 (como se mencionó en la sección 3.5, la placa tiene 54 entradas/salidas digitales, de las cuales 15 se pueden configurar como salidas PWM, siendo 9 y 10 dos de ellas).



Fig. 3.1.1: Analizador de Impedancias HP 4192A 5MHz-13MHz

En la figura 4.1.2 se muestran todas las bobinas medidas. En particular, elegimos la de 320 μH , 1.3 Ω a 31.25 kHz.

$$L = 320 \mu\text{H} \quad f = 31,25 \text{ kHz}$$

En la sección 3.6, vimos que el ripple de tensión a la salida del Buck es:

$$\Delta V_o = \frac{V_i * \delta * (1 - \delta)}{8 * C * L * f^2}$$

De la cual obtenemos que:

$$C = \frac{V_i * \delta * (1 - \delta)}{8 * \Delta V_o * L * f^2}$$

Además:

$$\delta = \frac{V_o}{V_i}$$

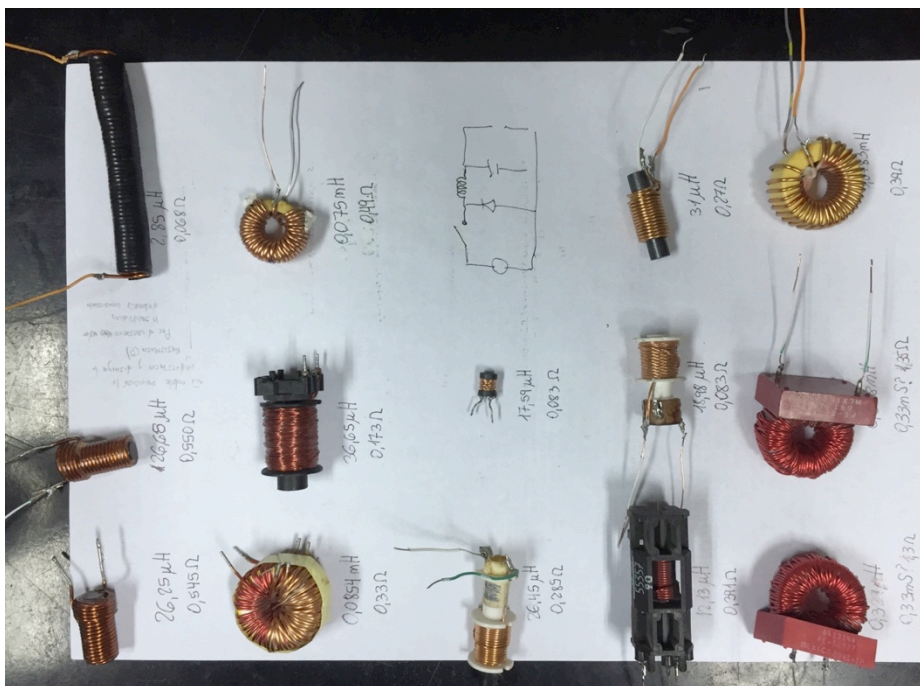


Fig. 3.1.2: Bobinas medidas con el analizador de impedancias

Para V_i utilizamos el valor de tensión de salida a la potencia máxima del panel fotovoltaico V_{pm} .

$$V_i = 17,5 \text{ V}$$

Para V_0 se considera 12 V, para utilizar el panel como cargador de baterías.

$$V_o = 12 \text{ V}$$

Entonces tenemos que:

$$\delta = 0,7$$

Para el ripple de tensión, se toma el %1 de la tensión de salida.

$$\Delta V_o = 120 \text{ mV}$$

Con todos estos datos obtenemos el valor del capacitor del Buck:

$$C = \frac{17,5 \text{ V} * 0,7 * (1 - 0,7)}{8 * 120 \text{ mV} * 320 \mu\text{H} * (31,25 \text{ kHz})^2}$$

$$C = 22,25 \mu\text{F}$$

Para la llave, utilizamos un MOSFET de potencia, el cual es capaz de manejar las tensiones y corrientes que entrega nuestro panel. En particular, decidimos utilizar un MOSFET de canal P para poder implementar el convertidor con la llave en el lado de arriba del circuito.

Además, debimos tener en cuenta que las salidas (y entradas) de la Arduino trabajan a 5 V – 20 mA. Por lo tanto, para poder polarizar el MOSFET era necesario elevar estos valores, en especial el de corriente. Con ese objetivo, decidimos agregar un transistor BJT que permitiera entregar la potencia requerida por el MOSFET, armando un circuito de polarización adecuado.

3.10.1 Simulaciones

Como mencionamos, en primer lugar, armamos un circuito de polarización básico para manejar el MOSFET, el cual se muestra en la siguiente figura:

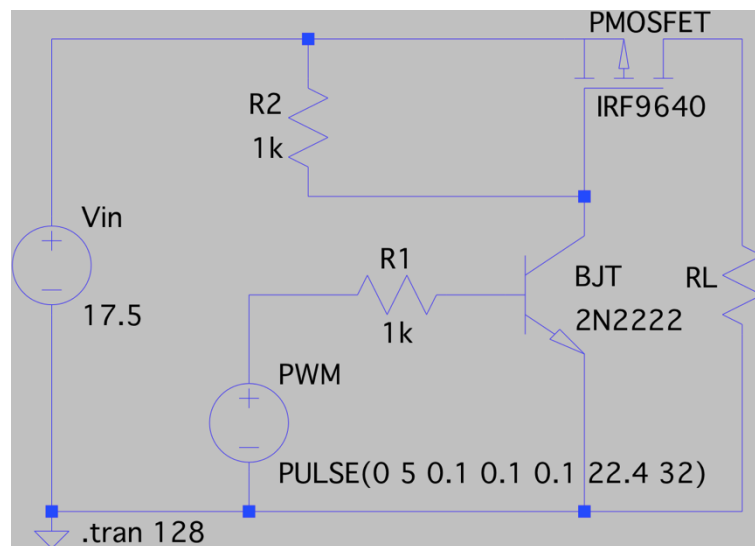


Fig. 3.2: Circuito de polarización

La señal PWM se conecta a la base del BJT mediante una resistencia de $1\text{k}\Omega$, y usamos otra resistencia del mismo valor para polarizar el MOSFET, entre el gate y el source del mismo.

Como tensión de salida (V_{RL}) obtuvimos:

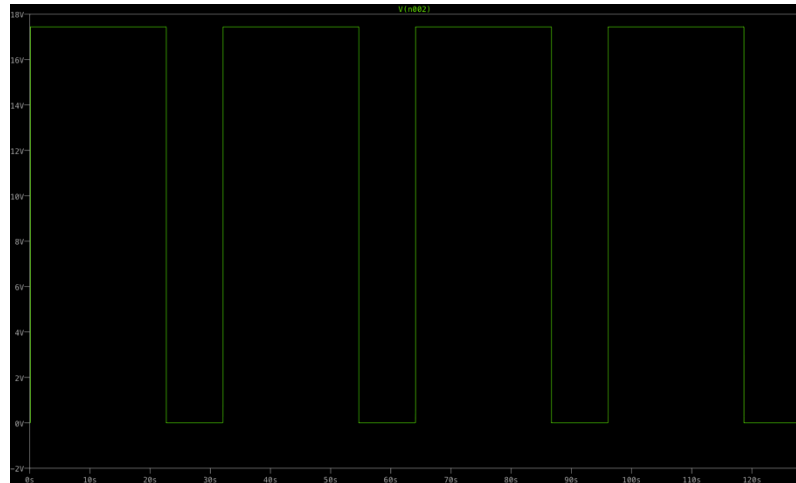


Fig. 3.3: Tensión a la salida del circuito de polarización

Como vemos, el circuito de polarización funciona correctamente, ya que la tensión a la salida se comporta como la tensión en el diodo de un convertidor tipo Buck. La PWM hace ver la tensión de entrada un 70% del tiempo, y el 30% restante cae a cero ($\delta = 0,7$).

Luego, combinamos este circuito con el resto del Buck:

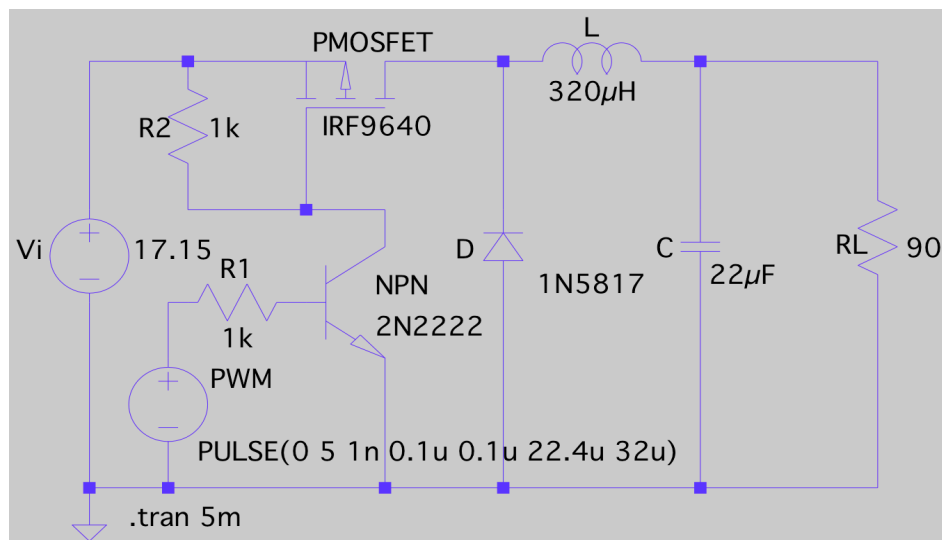


Fig. 3.4: Circuito del convertidor Buck completo

Como resistencia de carga utilizamos un valor de $90\ \Omega$, el cual es adecuado para representar una batería porque las mismas presentan resistencias bajas, cercanas a los $100\ \Omega$.

A la salida obtuvimos lo siguiente:

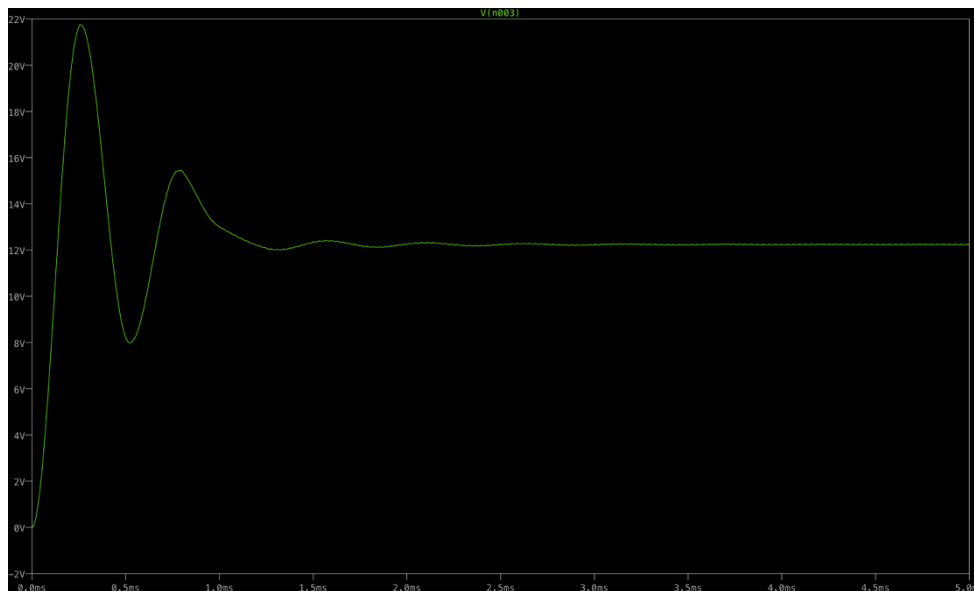


Fig. 3.5: Tensión de salida del Buck completo

Con lo cual se evidencia el correcto funcionamiento de nuestro circuito, ya que con 17,5 V de entrada se obtienen 12 V a la salida, tal como habíamos planteado en los cálculos teóricos para un $\delta = 0,7$.

3.11 Implementación algoritmo MPPT

Para la implementación de dicho algoritmo se usó el método de perturbar y observar el cuál se detalló anteriormente mediante el diagrama de flujo correspondiente.

Lo que se quería lograr era poder ver de manera gráfica como el algoritmo permitía ajustar el ciclo de trabajo del Buck que comanda una llave electrónica y así poder regular la potencia que se le envía a la carga. Siempre considerando que en todo instante dicho algoritmo intentará ajustar el ciclo para poder obtener la máxima potencia posible.

Entonces, primero se partió de poder obtener las curvas de corriente contra tensión y de potencia contra tensión para una determinada carga y en condiciones particulares de temperatura.

Las curvas se podían obtener de distintas formas mediante el cambio constante de carga y medir los parámetros de tensión y corriente o modificando el ciclo de trabajo del Buck pero para una determinada carga. Esto último fue lo que se realizó.

Las condiciones de temperatura fueron a 30°C aproximadamente, lo cual se logró mediante el acople a la estructura de tres ventiladores de fuentes de PC ya que de lo contrario la temperatura del panel se incrementaba constantemente lo que impediría obtener las curvas correctamente.

Los datos experimentales y los gráficos correspondientes a los mismos se detallan a continuación en una planilla de Excel.

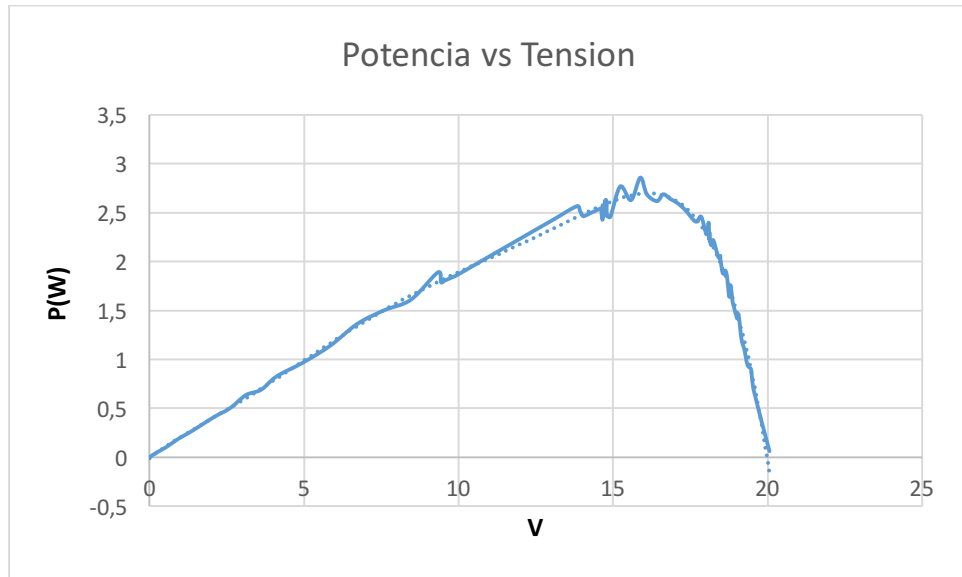


Fig. 3.11.1

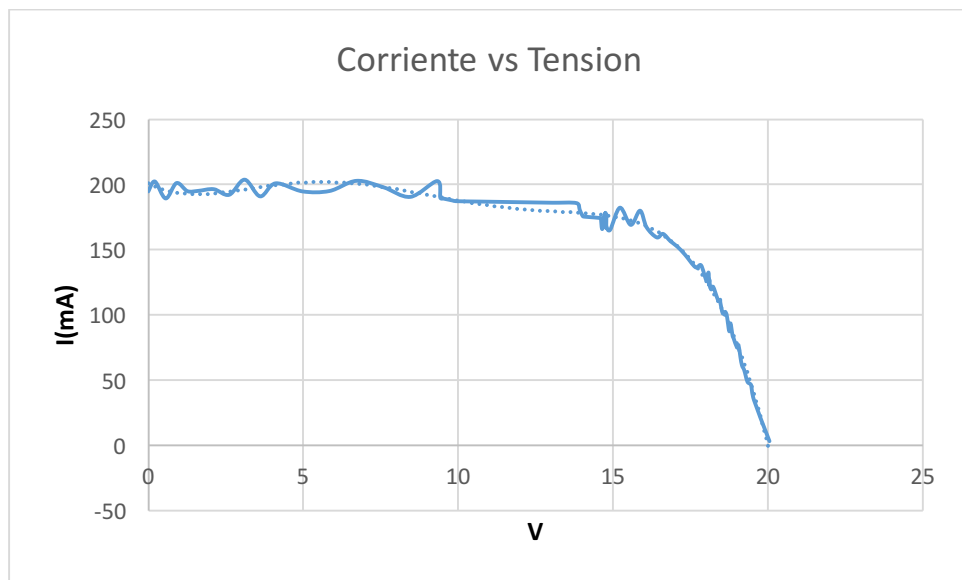


Fig. 3.11.2

Entonces de acuerdo a los datos experimentales, la potencia máxima que se obtuvo fue de **2,6 W** a una tensión y corriente de **17,08 V** y **152,4 mA**, respectivamente.

Por lo tanto, al momento de realizar esta parte del proyecto se debía llegar a los valores anteriores o cercanos y a una temperatura de 30°C.

El programa que se emplea para ejecutar el algoritmo se deja en la parte de anexos.

Una vez concluidas las pruebas previas para poder obtener una referencia de los valores a los que se debía llegar, se procedió a cargar el programa en la placa Arduino para posteriormente poder contrastar los datos obtenidos.

De esta manera lo que se obtuvo fue lo siguiente

```
228.21;17.70;128.90;143.00;29.25;  
220.06;17.63;124.80;140.46;29.14;  
228.50;17.67;129.30;137.92;29.25;  
213.93;17.77;120.40;140.08;29.25;  
225.20;17.68;127.40;142.24;29.25;  
221.01;17.65;125.20;139.70;29.57;
```

Fig.3.11.3: Monitor Serial Arduino

Lo que se observa en la pantalla es el monitor serial de Arduino el cual nos permite observar los datos numéricos que estábamos obteniendo en las distintas iteraciones del programa.

Los datos de la primera columna son la potencia que está multiplicada por un factor de 10 para poder después contrastar con la corriente que está en mili Ampere (mA). La segunda columna es la tensión, la tercera es el ciclo de trabajo, la cuarta es la corriente y finalmente la última columna es la temperatura.

Entonces observando estos datos se puede decir que a una temperatura de 29,57 °C se tuvo que la máxima potencia obtenida mediante la implementación del algoritmo fue de **2,21W** a una tensión y corriente de **17,65 V** y **140,24 mA**. El ciclo de trabajo en ese instante fue de 125,20 lo que representa (sabiendo que se trabaja con 8 bits por lo que 255 sería un 100% de ciclo de trabajo) aproximadamente un 49%.

Entonces se puede tener una primera conclusión de que el algoritmo se está ejecutando de acuerdo a los parámetros deseados en función de los datos obtenidos cuando se trazaron las curvas, esto porque comparando valores hay un aproximado de 400mV de diferencia y dicha diferencia puede ser porque las condiciones en las que se trazaron las curvas y las condiciones en las que se implementó el algoritmo no fueron las idénticas, por ejemplo, la posición del panel difiere de la posición original o la lámpara empleada no está en la misma posición.

Pero aún falta poder “ver” como es el comportamiento del algoritmo de manera gráfica ya que hasta ahora solo se observaron datos.

Para esto lo que se implementó fue el serial plotter de Arduino

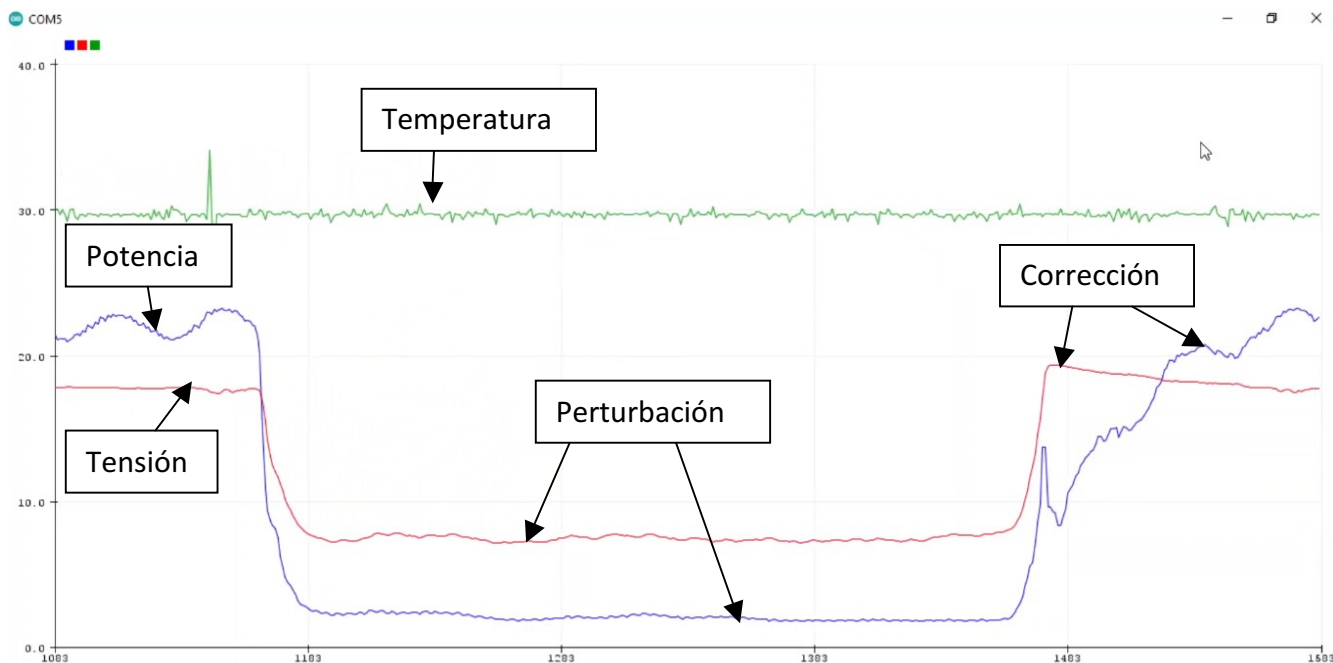


Fig.3.11.4: Serial Plotter

Del serial plotter se puede ver las gráficas de temperatura (verde), potencia (azul) y tensión (azul).

Lo que se hizo fue perturbar (colocar un papel, por ejemplo, sobre el panel) para observar cómo se comportaba el algoritmo. Al momento de perturbar se ve que la potencia y tensión bajan lo que quiere decir que el algoritmo corrige la potencia para “adecuarse” a esas condiciones y funcionar en una nueva curva.

Luego de retirar el papel, el algoritmo nuevamente corrige y vuelve a las condiciones iniciales, por lo tanto, esta es otra manera de poder ver cómo se comporta de manera dinámica el algoritmo en una gráfica.

3.12 Implementación algoritmo MPPT: Processing

Es un programa diseñado en lenguaje y creado u orientado para el diseño gráfico, animaciones y aplicaciones gráficas de todo tipo.

Desarrollado a partir del 2001 por el más que conocido departamento del MIT, el **Media Lab**. Los creadores, **Casey Res y Ben Fry**, son discípulos del profesor **John Maeda creador del método DBN** (Design By Numbers) que, en los años 90, pretendía introducir a la programación a diseñadores, artistas y no programadores de una forma sencilla. Precisamente Processing está inspirado en DBN que en la actualidad ya no está activo.

Processing es un lenguaje **plug and play** es decir, no es necesario instalar cosas o configurar ya que con la sola instalación es posible empezar a programar.

Existen 3 maneras de programar en esta plataforma:

- De forma básica, tipo Basic o Ensamblador. Sentencia a sentencia, con variables globales y sin nada de complejidad.

- Podemos utilizar la programación procedural o estructurada como en C. Algo más compleja pero mucho más limpia donde crearemos nuestras propias funciones a las que llamaremos.
- También podemos utilizar la manera más compleja, utilizando toda la potencia de la programación orientada a objetos.

Cuando estamos programando una aplicación de este estilo, sea cual sea el objetivo, lo que buscamos es poder **ejecutar nuestro proyecto en cualquier ordenador**. Processing nos da la posibilidad de **generar un ejecutable para las diferentes plataformas** Mac OS, Windows o Linux e incluso podemos desarrollar **aplicaciones móviles gracias a la SDK** que nos ofrecen para Android. No hay que olvidar que este sistema operativo móvil está basado en Java.

Por último, aunque no menos importante, podemos conectar Processing con Arduino ya que están estrechamente ligados.

Esto último es porque, su IDE (*Integrated Development Environment o Entorno de Desarrollo Integrado*) es muy similar al de Arduino ya que la forma en que se programa tiene la misma estructura que este último, ya que tiene un espacio para definir variables globales, tiene un void setup y un void loop aunque a este último lo define como void draw pero sigue siendo una parte del programa que se ejecuta constantemente.

Además, se permite la comunicación bidireccional entre Arduino y Processing habilitando los puertos que correspondan.

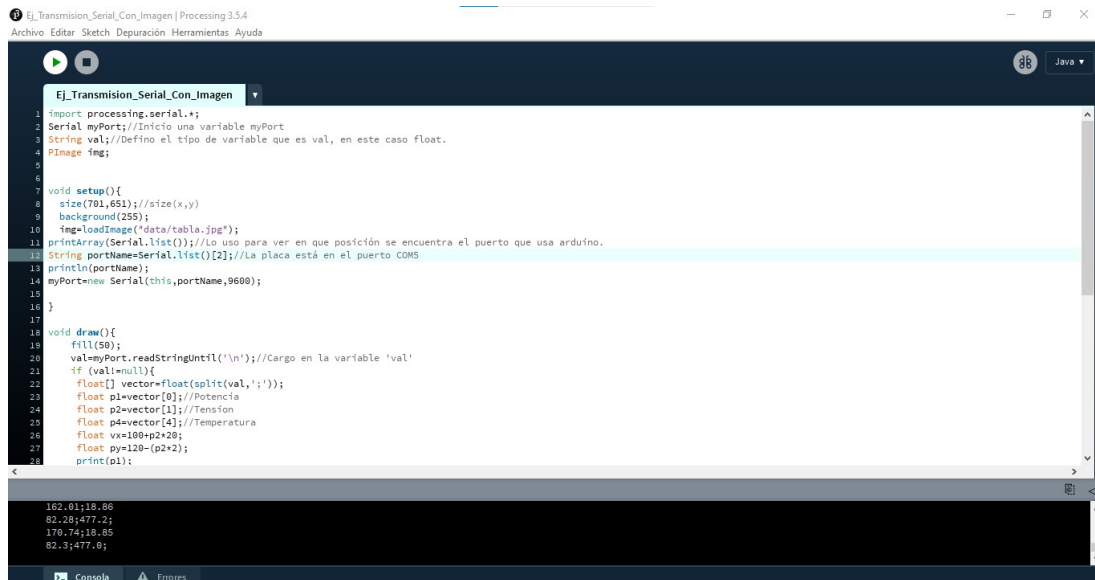


Figure 1 Ejemplo de lo que sería la estructura de un programa dentro del IDE de Processing

3.12.2 Arduino y Processing

En esta parte del proyecto lo que se hizo fue utilizar Processing como entorno gráfico para poder ver las curvas del mppt y observar cómo se desplazaba el punto de máxima potencia como consecuencia del algoritmo.

Esto ya que en Arduino podíamos ver como se mantenía constante la potencia en función del tiempo, pero no teníamos manera de contrastarla con las curvas ideales o mejor dicho las que se obtuvieron experimentalmente variando el ciclo de trabajo del Buck bajo las consideraciones de temperatura constante a 30 °C.

Como se comentó anteriormente, lo que se quería lograr era poder ver las curvas de corriente versus tensión y la de potencia versus tensión, todas sobre la misma imagen. Además, poder leer los datos de potencia, tensión (productos del algoritmo mppt) y temperatura proveniente del Arduino para así poder plasmarlos de manera gráfica.

La idea era poder contrastar los datos de potencia y tensión producto del algoritmo mppt con las curvas detalladas anteriormente. Estos datos del algoritmo se verían como un punto que constantemente se posicionaría alrededor de la zona considerada de máxima potencia, pero surgió un problema ya que al intentar hacer esto los datos antiguos de potencia versus tensión persistían sobre la imagen por lo que se optó por graficar constantemente las curvas lo que solucionaba el problema del punto que no se borraba, pero en cada ciclo solo se obtenía un trozo de dichas curvas.

Por lo tanto, como solución final se optó por sacar una copia de la imagen y guardarla en sus coordenadas originales de manera tal que no se modificara su escala y luego cargarla por programa. Esta fue la solución que mejor se adaptó a las necesidades anteriores por lo que se solucionó el problema de la persistencia del punto y de las gráficas de las curvas que se veían sin uniformidad.



Figure 2 Graficas que muestran los rastros del punto como consecuencia de su desplazamiento debido a los ajustes que hace el algoritmo mppt



Figure 3 Gráficas con los ajustes necesarios para que no se observen los rastros que deja el punto como consecuencia del ajuste que hace el algoritmo



Figure 4 Grafica que muestra el desplazamiento del punto respecto a la anterior gráfica como consecuencia de una perturbación

4 Anexos

4.1 Diseño de la estructura y engranajes

Para la estructura que sostiene al panel y los servomotores se utilizó una placa de fibrofácil MDF de 9 mm ya que es más fácil de trabajar al momento de realizar los cortes mientras que para los engranes se utilizó filamento PLA 1.75 mm para su posterior fabricación en una impresora 3D.

Si bien el diseño estructural no fue difícil se tuvo que tener en cuenta que las dimensiones de los cortes no debían ser superficies grandes ya que, teniendo en cuenta al panel, agregaría un peso mayor al que podría soportar el servo azimuth debido a que sobre este se encuentra la mayor parte de la estructura.

Por otro lado, para que los servos no se vean demasiado forzados al momento de realizar los movimientos de azimuth y vertical se diseñaron unos engranes, que, de manera teórica, se pudo hacer al engrane conductor (el que está sobre el eje del servo) lo más chico posible respecto al conducido (el que está fijo al panel y el que está fijo a la estructura) para que el torque de salida sea lo mayor posible respecto al torque de entrada (que es el que ejerce el servo).

Pero como se dijo anteriormente, esto en forma teórica, ya que el engrane conductor no se puede hacer demasiado pequeño debido a la estructura del servo porque el engrane conducido puede chocar con dicha superficie y limitar de manera perjudicial el giro de los engranes.

Entonces, una vez explicadas las consideraciones que se tuvieron al momento de realizar el proyecto, se procederá a explicar el proceso que se llevó a cabo para la realización de la estructura y de los engranes.

4.1.1 Geometría de engranes.

Los engranes se diseñaron teniendo en cuenta que lo que se quería tener era un torque de salida lo más grande respecto del torque de entrada. Para ello se emplearon las ecuaciones de cálculos de engranes que están basadas en su geometría y en el par que pueden generar.

$$m = \frac{dp}{Z}$$

- Módulo (m). Es la relación entre el diámetro primitivo y el número de dientes. Para que dos engranes puedan engranar deben tener el mismo módulo.
- Diámetro primitivo (dp). Es el diámetro de la circunferencia primitiva y está dado en milímetros.
- Número de dientes (Z) que contiene un engrane en particular.

Para poder seguir con el desarrollo es conveniente describir otras partes que componen la geometría y estructura de un diente.

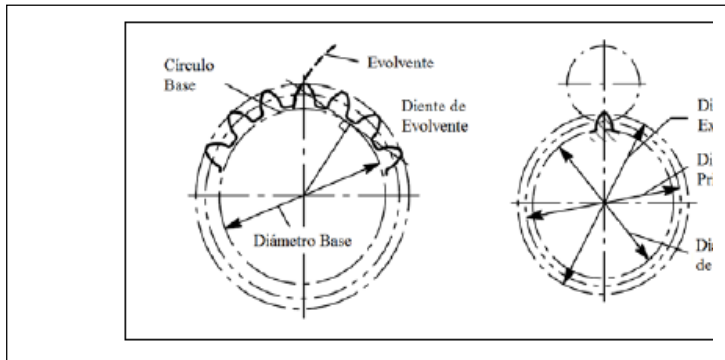


Fig. 4.2.1

- Adendo o altura de cabeza (h_a). Es la distancia radial entre la circunferencia primitiva y la cabeza del diente. Para un perfil de referencia normalizado $h_a = m$

- Dedendo o altura de pie (h_f). Es la distancia radial entre la raíz del diente y la circunferencia primitiva. Para un perfil de referencia normalizado, $h_f = 1,25 * m$

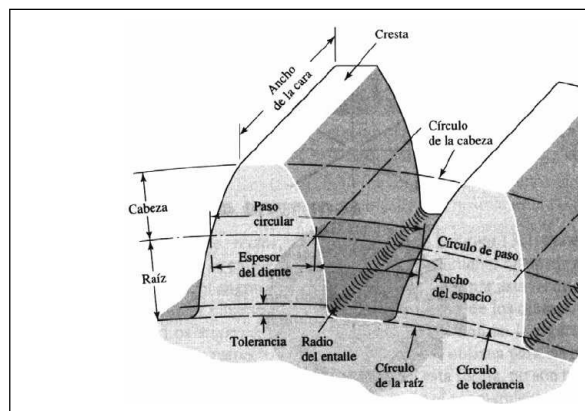


Fig. 4.2.2

- Paso (p). También llamado paso circular o circunferencial es la distancia medida sobre la circunferencia primitiva entre puntos homólogos de dos dientes consecutivos. De acuerdo a la figura sería la suma del grueso del diente y el ancho entre dos dientes consecutivos o se puede calcular como.

$$p = \frac{\pi * dp}{Z} = \pi * m$$

- Radio de entalle (r). Es el radio de la cuna de empotramiento del diente en el engrane. Se calcula como

$$r = \frac{\pi * m}{12}$$

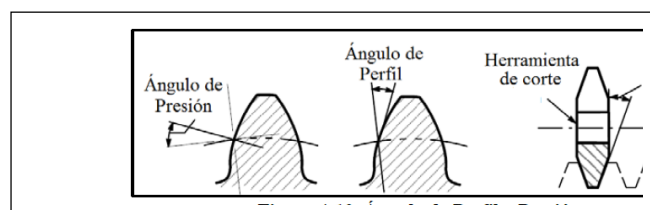


Fig. 4.2.3

- Angulo de presión (α). Es el ángulo que forma la línea de presión (normal a la superficie del diente en el punto de contacto entre dos engranes) con la tangente a ambas circunferencias primitivas. Este ángulo se puede determinar como

$$r_b = r_p * \cos (\alpha)$$

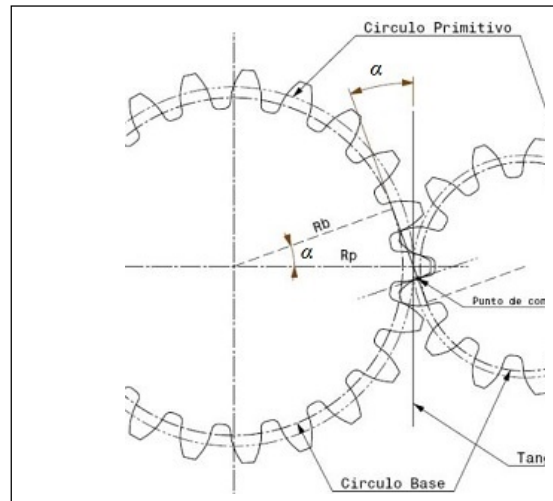


Fig. 4.2.4

Donde r_b es el radio base y r_p es el diámetro primitivo. Los valores más utilizados son de 20° , $22 \frac{1}{2}^\circ$ y 25° , aunque alguna vez se utilizó $14 \frac{1}{2}^\circ$.

- Angulo entre dientes (θ). Se define como

$$\theta = \frac{360^\circ}{Z}$$

4.1.2 Par de fuerzas y potencia transmitida

La expresión matemática que define la potencia que transmite un eje de transmisión en función del par de fuerzas y de su velocidad angular de giro, es la siguiente:

$$P = \tau \cdot \omega$$

siendo,

P , la potencia transmitida por el eje, en W (vatios);

τ , es el par de fuerzas que desarrolla, en $N \cdot m$;

ω , es la velocidad angular a la que gira el eje, en rad/s .

La anterior expresión es básica y permite entender las prestaciones de una transmisión por engranajes.

En efecto, suponiendo una transmisión con un eje de entrada del movimiento (*Eje 1*) y un eje de salida (*Eje 2*), y que la potencia de entrada (P_1) y la de salida (P_2) sean iguales al considerarse nulas las pérdidas que se puedan producir en la transmisión (*rendimiento = 1*), se cumpliría lo siguiente:

$$P_1 = \tau_1 \cdot \omega_1$$

$$P_2 = \tau_2 \cdot \omega_2$$

Al ser ambas potencias iguales, se tiene que:

$$P_1 = P_2$$

$$\tau_1 \cdot \omega_1 = \tau_2 \cdot \omega_2$$

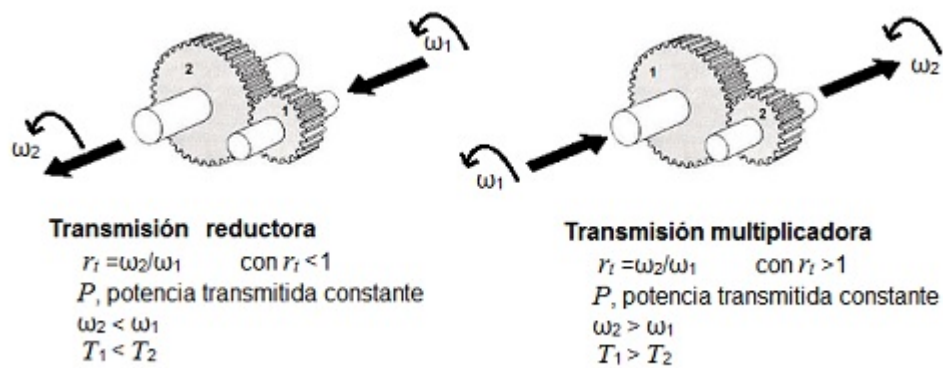


Fig.4.3.1 Tipos de transmisiones

De esta manera, cuando se emplea una transmisión reductora ($r_t = \omega_2 / \omega_1$ con $r_t < 1$), donde la velocidad de giro del eje de salida es menor que la de entrada ($\omega_2 < \omega_1$), y al conservarse constante el producto par de fuerza por velocidad de giro ($\tau_1 \cdot \omega_1 = \tau_2 \cdot \omega_2$), se consigue un par a la salida de la transmisión (τ_1) mayor que el de entrada ($\tau_2 > \tau_1$).

Y a la inversa, en una transmisión multiplicadora ($r_t = \omega_2 / \omega_1$ con $r_t > 1$), donde la velocidad de giro del eje de salida es mayor que la de entrada ($\omega_2 > \omega_1$), se tiene un par de fuerzas a la salida de la transmisión (τ_2) menor que el par que desarrolla el eje de entrada ($\tau_2 < \tau_1$).

4.1.3 Cálculo de engranes

Como se dijo anteriormente, para que dos engranes puedan engranar se es necesario que tengan igual módulo

$$m_1 = m_2$$

$$\frac{dp_1}{Z_1} = \frac{dp_2}{Z_2}$$

Por lo tanto

$$\frac{Z_2}{Z_1} = \frac{dp_2}{dp_1} = m$$

De esta manera se puede obtener el módulo como una relación entre los dientes de los engranes. Entonces, por convención, se puede considerar de manera óptima un módulo igual a 3 o lo que es lo mismo, que el número de dientes del engrane conducido tiene tres veces más el número de dientes del engrane conductor y en consecuencia el esfuerzo que deberá hacer el servo será mucho menor respecto al caso en que el módulo hubiera sido menor.

Pero esto sería en principio una consideración deseable ya que, como se especificó anteriormente, el soporte que se encuentra en el eje del servo y al cuál se acopla el engrane conductor tiene un diámetro determinado por lo que el diámetro del engrane no podrá ser menor. El diámetro del soporte del engrane es aproximadamente 22 mm.

Entonces, suponiendo un $Z_1 = 13$ y un $m = 3$ se obtiene un $Z_2 = 39$. Por lo tanto

$$\begin{aligned} dp_1 &= m * Z_1 = 3 * 13 \text{ mm} \\ dp_1 &= 39 \text{ mm} \\ dp_2 &= m * Z_2 = 3 * 39 \text{ mm} \\ dp_2 &= 117 \text{ mm} \end{aligned}$$

En principio estas serían algunas de las medidas y características de los engranes, pero hay que seguir obteniendo más datos para poder obtener conclusiones.

$$\begin{aligned} h_a &= m \text{ (altura de cabeza)} \\ h_a &= 3 \text{ mm} \\ h_f &= 1,25 * m \text{ (altura de pie)} \\ h_f &= 3,75 \text{ mm} \\ d_f &= \text{Diámetro de pie} \\ d_f &= d_p - 2 * h_f \\ d_{f1} &= 39 \text{ mm} - 2 * 3,75 \text{ mm} \\ d_{f1} &= 31,5 \text{ mm} \end{aligned}$$

d_{f1} es del engrane conductor y debe ser mayor que el diámetro del soporte (22 mm). Por lo tanto, se cumple la condición y en principio se podría emplear un engrane con dichas características. Siguiendo con el procedimiento se obtiene

$$\begin{aligned} d_h &= \text{Diámetro de cabeza} \\ d_h &= d_p + 2 * h_a \\ d_{f1} &= 39 \text{ mm} + 2 * 3 \text{ mm} \\ d_{f1} &= 33 \text{ mm} \end{aligned}$$

Antes de seguir con los cálculos, se tuvo que tener en cuenta que el servo solo puede girar 180° por ende se debe considerar cuanto girará el engrane conducido en función de lo que gire el engrane conductor.

Se sabe que el movimiento vertical del panel debe ser como mínimo de 0° a 90° para poder captar la radiación solar lo mejor posible, entonces el engrane conducido

debe girar 90° cuando el engrane conductor gire 180°. De la función de transferencia para engranes se obtiene

$$Z_1 * \theta_1 = Z_2 * \theta_2$$

$$\frac{Z_2}{Z_1} = \frac{\theta_1}{\theta_2} = m$$

Si θ_1 es el giro en grados del engrane acoplado al servo entonces

$$\theta_2 = \frac{\theta_1}{m}$$

$$\theta_2 = \frac{180^\circ}{3}$$

$$\theta_2 = 60^\circ$$

Quiere decir que, en las condiciones actuales, cuando el engrane conductor gire 180° el conducido solo hará 60°, es decir que nunca alcanzará los 90° requeridos por ende hay que modificar el módulo al aparecer este limitante.

El módulo ahora se cambia a $m = 2$ y de esta manera θ_2 puede llegar a 90°. Recalculando todo nuevamente con $Z_1 = 14$ para que el diámetro de pie sea mayor al del soporte que permite acoplar el engrane al servo

$$dp_1 = 28 \text{ mm}$$

$$Z_2 = 28$$

$$dp_2 = 56 \text{ mm}$$

$$h_a = 2 \text{ mm}$$

$$h_f = 2,5 \text{ mm}$$

$$d_{f1} = 23 \text{ mm}$$

$$d_{f2} = 51 \text{ mm}$$

$$d_{a1} = 32 \text{ mm}$$

$$d_{a2} = 60 \text{ mm}$$

Otros datos necesarios para el diseño de los engranes

$$p = \frac{\pi * dp}{Z} = \pi * m \text{ (paso)}$$

$$p = 6,2832$$

$$r = \frac{\pi * m}{12} \text{ (radio de entalle)}$$

$$r = 0.5236$$

$$\theta = \frac{360^\circ}{Z} \text{ (angulo entre dientes)}$$

$$\theta_1 = 25,7143^\circ$$

$$\theta_2 = 12,857^\circ$$

Además se utilizó un ángulo de presión $\alpha = 25^\circ$ como una primera consideración ya que primero se diseña con el software Solidworks el cual permite ver tridimensionalmente los engranes y como es el contacto entre los mismos.

El resultado del diseño en el software Solidworks luego de colocar los diferentes parámetros calculados es

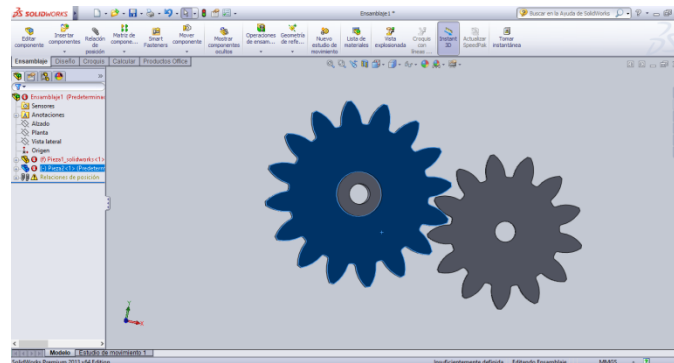


Fig.4.4.1

En la imagen se puede observar que el engrane entre dientes es bastante bueno por lo que el considerar un $\alpha = 25^\circ$ está bien.

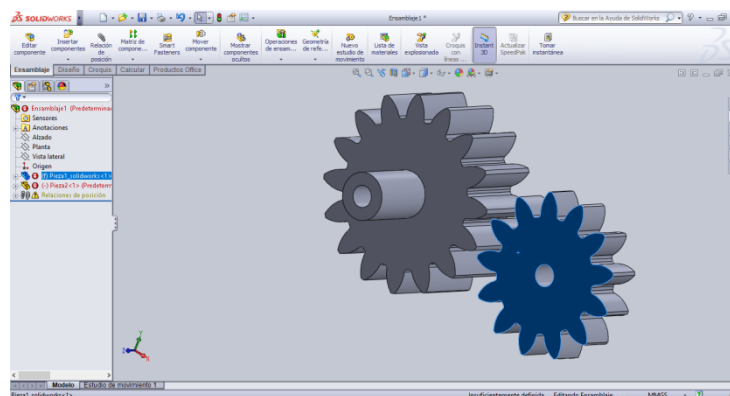


Fig. 4.4.2

Luego de ver el modelo 3D de los engranes se procedió a cambiar el formato de los archivos de .SLDASM a .STL o para su posterior impresión en la impresora de la sala de proyectos de la facultad.

4.2 Programa principal

```
int bot_mppt=3;//No sé si será necesario agregar const.
int bot_seguidor=2;
int val_mppt=0;
int val_seguidor=0;

//MPPT
#include <Wire.h>
#include <Adafruit_INA219.h>
Adafruit_INA219 ina219;
    // Declaracion de variables globales del sensor LM35
float tempC; // Variable para almacenar el valor obtenido del sensor
(0 a 1023)
int pinLM35 = 6; // Variable del pin de entrada del sensor (A0)
unsigned long tiempo1 = 0;//Variable para medir el tiempo que pasa
entre cada medición de temperatura.

    // Declaración variables para mediciones con el INA219 y
Configuración DC del Buck
int PinPWM=10; //Defino la salida PWM en el Pin 10
float D;
float Dant=127; //Comienzo con un ciclo de trabajo del 50%
float deltaD=2.54; //Cambio en el ciclo de trabajo de 1%=2.54
float Vant=14.0;
float Pant=1000e-3;
int PantG;
int PG;
float I;
float V;
float P;
float deltaPG;
float deltaV;
float SV;
float BV;
//Fin_MPPT

//Seguidor
#include<Servo.h>
Servo servol;
int LDRah;//Es el que indica si el giro es antihorario.
int LDRh;//Es el que indica si el giro es horario.
float angl=60;
float angdelta=1;//Incremento o decremento el angulo de a 1°.
float dif1=0;
Servo servo2;
int LDRder;//Es el que indica si el giro hacia derecha.
int LDRiz;//Es el que indica si el giro es izquierda.
float ang2=45;
float dif2=0;
//Fin_Seguidor

void setup() {
uint32_t currentFrequency;
Serial.begin(9600); //Establece la tasa de baudios
ina219.begin(); //Inicia la biblioteca para el INA219
TCCR2B = TCCR2B & 0b000 | 0x01; //Cambia la frecuencia de la salida
PWM del Arduino a 31.25 kHz
pinMode(bot_mppt, INPUT);
pinMode(bot_seguidor, INPUT);
```

```

//Seguidor
servo1.write(8); //Al hacer esto le doy una posición inicial al
servo y así no se autoposiciona automáticamente a 90 grados.
servo1.attach(5);
Serial.println("Angulo inicial Servo Vertical:");
Serial.println(90);
for (int i = 0; i <= 90; i++)
{
    servo1.write(i);
    delay(100);
}
servo2.write(8); //Angulo inicial
servo2.attach(6); //
Serial.println("Angulo inicial Servo Azimut:");
Serial.println(8);
for (int j = 0; j <= 45; j++)
{
    servo2.write(j);
    delay(100);
}
//Fin_Seguidor
}

void mppt() { //Subrutina que ejecuta
el control mppt
SV=ina219.getShuntVoltage_mV();
BV=ina219.getBusVoltage_V();
I=ina219.getCurrent_mA();
V=BV+(SV/1000);
P=(V*I)/1000;
PantG=Pant*1000; //antes era 1000
//Serial.print(PantG); Serial.println("");
PG=P*1000; //antes era 1000
//Serial.print(PG); Serial.println("");
deltaV=V-Vant;
deltaPG=PG-PantG;
//Serial.print("deltaV: ");
//Serial.print(deltaV);
//Serial.print(" ");
//Serial.print("deltaPG: ");
//Serial.print(deltaPG);
//Serial.println("");
//delay(1000);
if (deltaPG!=0) {
    if(deltaPG>0) {
        if(deltaV>0){
            D=Dant-deltaD;
        }
        else {
            D=Dant+0.85*deltaD; // En los incrementos hace esto
0.85*deltaD;
        }
    }
    else {
        if(deltaV>0){
            D=Dant+0.85*deltaD;

```

```

    }
    else {
        D=Dant-deltaD;
    }
}
}
else{
    D=Dant;
}

if (D>241.3) {
    D=203.2;
}
if (D<12.7) {
    D=12.7;
}
//Serial.print("Vant: "); Serial.print(Vant); Serial.println(" V");
//Serial.print("Pant: "); Serial.print(Pant*1000); Serial.println("
mW");
//Serial.print("Dant: "); Serial.print(Dant/2.54);
Serial.println("");
//Serial.println("");
    Dant=D; //Esto estaba en el original
    Vant=V;
    Pant=P;
analogWrite(PinPWM,D);
Serial.print(100*P); //comando para graficar, lo multiplico por 100
para poder escalarlo mejor y poder verlo en processing..
Serial.print(";");
Serial.print(V);
Serial.print(";");
Serial.print(I); //mA
Serial.print(";");
Serial.print(D);
Serial.print(";");

//INSTRUCCIONES PARA MEDIR TEMPERATURA
    tempC = analogRead(pinLM35);
    tempC = (1.1 * tempC * 100.0)/1023; // Calculamos la temperatura
con la fórmula
    //Serial.print("La temperatura actual es:");
    Serial.print(tempC); // Envia el dato al puerto serial. Para verlo
en la gráfica debo agregar Serial.println
    //Serial.println(" °C ");
    Serial.println(";");
    delay(800);
//Al variar el valor inicial del ciclo de trabajo, la tensión máxima
que entrega la PWM varía
}

void seguidor() { //Subrutina que ejecuta el
seguidor solar.
    LDRah=analogRead(0); //Cable negro.
    LDRh=analogRead(1); //Cable azul.
    dif1=LDRh-LDRah;
    Serial.println(dif1);
    Serial.println(servo1.read());

```

```

dif1=LDRh-LDRah;
Serial.println(dif1);
Serial.println(servo1.read());
//delay(10);
if(dif1<=20 & -20<=dif1)//Rangos (dif1<=31 & -30<=dif1)
{
    servo1.write(ang1);
}
else
{
    if (dif1<-20 & ang1>10)//Mayor tensión cable
negro.Decremento el angulo,sentido antihorario
    {
        Serial.print("dif1 antihorario ");Serial.println(dif1);
        ang1=ang1-angdelta;
        Serial.println(ang1);
        servo1.write(ang1);
    }
    else if (ang1<=10)//Límite inferior de giro del servo.
    {
        Serial.println("Angulo minimo:");
        Serial.println(ang1);
        servo1.write(11);
        //angmant1=ang1;
    }
}

    if (dif1>20 & ang1<92)
    {
        Serial.print("dif1 horario ");Serial.println(dif1);
        ang1=ang1+angdelta;
        servo1.write(ang1);
    }
    else if (ang1>=92)//Limte superior de giro del servo.
    {
        servo1.write(91);
        //angmant1=ang1;
    }
}

delay(10);//Aquí comienza las instrucciones para el servo Azimut.
LDRiz=analogRead(9);//cable violeta//.
LDRder=analogRead(8);//cable negro.
dif2=LDRiz-LDRder;
Serial.println(dif2);
if(dif2<=10&dif2>=-10)//Rangos (dif1<=31 & -30<=dif1)
{
    servo2.write(ang2);
    Serial.println("mantengo");
}
else {
    if (dif2<-10 & ang2>15)//Tension cable negro es mayor.
    {
        ang2=ang2-angdelta;
        servo2.write(ang2);
        Serial.println(ang2);
        Serial.println("giro derecha");
    }
    else if (ang2<=15)//Ver con respecto al giro
de los ldr.

```

```

        {
            servo2.write(16);
        }
    }

    if (dif2>10 & ang2<95)//Tension cable azul mayor.
    {
        Serial.println(ang2);
        Serial.println("giro izquierda");
        ang2=ang2+angdelta;
        servo2.write(ang2);
    }
    else if (ang2>=95)//Limite superior de giro del
Servo Azimut.
    {
        servo2.write(94);
    }

    delay(10);
}

void loop() {
    val_mppt=digitalRead(bot_mppt);
    Serial.print("bot_mppt= ");Serial.println(val_mppt);
    delay(10);

    val_seguidor=digitalRead(bot_seguidor);
    Serial.print("bot_seguidor= ");Serial.println(val_seguidor);
    delay(10);

    if (val_mppt==HIGH & val_seguidor==LOW){
        Serial.println("Rutina mppt");
        mppt();
    }
    else if (val_seguidor==HIGH & val_mppt==LOW){
        Serial.println("Rutina Seguidor");
        seguidor();
    }
    else {
        Serial.println("Rutina nada");
        delay(10);
    }
}
}

```

5 Bibliografía

- Basic in Solar Radiation at Earth Surface – Lucien Wald
- Solar Enginnering of Thermal Processes – 4° Ed.- Jhonn A. Duffie, William A. Beckman
- Ingeniería de Control Moderna – 5° Ed.-KatsuhikoOgata
- Sistemas de Control Automático – 7° Ed.- Benjamin C. Kuo
- <https://www.arduino.cc/>
- <https://processing.org/>
- Teoría de Electrónica Industrial
- Teoría de Instrumentos y Mediciones II
- Teoría de Control I

6 Conclusiones

En términos generales se logró llevar cabo el proyecto logrando concordancia entre los desarrollos teóricos y la implementación práctica.

De esta manera pudimos comprender mejor los aspectos básicos sobre el funcionamiento de un seguidor solar y como se puede complementar con el algoritmo mppt.

A pesar de que se lograron los objetivos básicos queda detallar algunos aspectos que no se tuvieron en cuenta o problemas que surgieron durante el desarrollo del proyecto:

- No se linealizó la respuesta de los LDR ya que su curva de tensión en función de la iluminación es no lineal, de manera tal que si se hubiera hecho se podría mejorar la respuesta ante variaciones de luz y se mejoraría el posicionamiento del panel, pero esto implicaría agregar circuitería externa.
- Al implementar el algoritmo debido al programa que emplea Arduino se tenía un problema al modificar el ciclo de trabajo ya que cerca de los valores extremos dicho programa hacía que se fuera de los rangos permitidos e hiciera caso omiso a las instrucciones dadas por el algoritmo. De esta manera lo que se hizo fue limitar el ciclo de trabajo por lo que ya no podía tomar cualquier rango de valores.
- En principio se compró una fuente reductora Buck suponiendo que sería una ventaja ya que permitía conectarse directamente con Arduino, pero no se consideró que su ciclo de trabajo solo se podía modificar mediante un potenciómetro externo propio de la placa y esto no nos permitía tener un control de la PWM mediante alguna de las salidas digitales de Arduino. Este fue el motivo por el cual decidimos armar una placa propia.

Para finalizar, se puede considerar a este proyecto como complementario a otros como lo pueden ser un regulador de carga y un circuito inversor y de esta manera tener en detalle lo que sería una conexión eléctrica domiciliaria, por ejemplo.