

Centro Nacional de Investigación y Desarrollo Tecnológico

Departamento de Ingeniería Electrónica

TESIS DE MAESTRÍA EN CIENCIAS

Controlador Basado En FPGA para Sistemas Fotovoltaicos

presentada por

Ing. Rita Ávila Medina

como requisito para la obtención del grado de
Maestra en Ciencias en Ingeniería Electrónica

Director de tesis

Dr. Jorge Hugo Calleja Gjumlich

Codirector de tesis

Dr. Carlos Aguilar Castillo

Cuernavaca, Morelos, México. Julio de 2015.

"2015, Año del Generalísimo José María Morelos y Pavón"

Cuernavaca, Mor., 26/Junio/2015

OFICIO No. DIE/102/2015

Asunto: Aceptación de documento de tesis

DR. GERARDO V. GUERRERO RAMÍREZ
SUBDIRECTOR ACADÉMICO
PRESENTE

Por este conducto, los integrantes de Comité Tutorial de la C. **Rita Ávila Medina**, con número de control M13CE002, de la Maestría en Ciencias en Ingeniería Electrónica, le informamos que hemos revisado el trabajo de tesis profesional titulado "**CONTROLADOR BASADO EN FPGA PARA SISTEMAS FOTOVOLTAICOS**", y hemos encontrado que se han realizado todas las correcciones y observaciones que se le indicaron, por lo que hemos acordado aceptar el documento de tesis y le solicitamos la autorización de impresión definitiva.

DIRECTOR DE TESIS



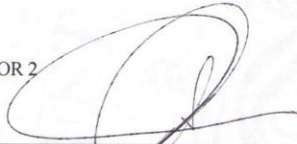
Dr. Jorge Hugo Calleja Gjumlich
Doctor en Ciencias en Ingeniería Electrónica
Cédula Profesional 4033213

REVISOR 1



Dr. Jesús Dario Mina Antonio
Doctor en Ingeniería
Cédula Profesional 6094069

REVISOR 2



Dr. Marco Antonio Oliver Salazar
Doctor en Ciencias en Ingeniería Electrónica
Cédula Profesional 6526083

REVISOR 3



Dr. Carlos Aguilar Castillo
Doctor en Ciencias en Ingeniería Electrónica
Cédula Profesional 3581040

C.p. L.I. Guadalupe Garrido Rivera.- Jefa del Departamento de Servicios Escolares.
Estudiante
Expediente

CMAZ/lrr

"2015, Año del Generalísimo José María Morelos y Pavón"

Cuernavaca, Mor., 29 de junio de 2015
OFICIO No. SAC/167/2015

Asunto: Autorización de impresión de tesis

**ING. RITA ÁVILA MEDINA
CANDIDATA AL GRADO DE MAESTRA EN CIENCIAS
EN INGENIERÍA ELECTRÓNICA
PRESENTE**

Por este conducto, tengo el agrado de comunicarle que el Comité Tutorial asignado a su trabajo de tesis titulado "**Controlador Basado en FPGA para Sistemas Fotovoltaicos**", ha informado a esta Subdirección Académica, que están de acuerdo con el trabajo presentado. Por lo anterior, se le autoriza a que proceda con la impresión definitiva de su trabajo de tesis.

Esperando que el logro del mismo sea acorde con sus aspiraciones profesionales, reciba un cordial saludo.

ATENTAMENTE
"CONOCIMIENTO Y TECNOLOGÍA AL SERVICIO DE MÉXICO"



**DR. GERARDO VICENTE GUERRERO RAMÍREZ
SUBDIRECTOR ACADÉMICO**



CENTRO NACIONAL DE
INVESTIGACIÓN Y
DESARROLLO
TECNOLÓGICO
SUBDIRECCIÓN
ACADÉMICA

C.p. Lic. Guadalupe Garrido Rivera.- Jefa del Departamento de Servicios Escolares.
Expediente

GVGR/mcr

Dedicatoria:

A Dios por su amor, fidelidad, puedo decir libremente "Eben-ezer" (Hasta aquí me ayudó Jehová 1 Samuel 7:12).

A mi Madre: Rosy Medina por su amor y apoyo incondicional.

A mis hermanos: Paco Avila e Isaac Avila por darme siempre ánimo para seguir adelante y poner su confianza en mí.

*"Bendice, alma mía, a Jehová,
¡y bendiga todo mi ser su santo nombre.
Bendice, alma mía, a Jehová,
¡y no olvides ninguno de sus beneficios.
Él es quien perdona todas tus iniquidades,
Él que sana todas tus dolencias;
Él que rescata del hoyo tu vida,
Él que te corona de favores y misericordias;
Él que sacia de bien tu boca
De modo que te rejuvenezcas como el águila".*

Salmo 103:1-5

¡¡Aleluya!!

Agradecimientos:

"Al único y sabio Dios, nuestro Salvador, sea gloria y majestad, imperio y potencia, ahora y por todos los siglos. Amén".

Judas 1:25

A Dios por su fidelidad, por su dirección, sus promesas, su amor, su apoyo, faltarían paginas para agradecer todo lo que ha hecho por mi tanto en lo espiritual, personal, y académico, a ti sea la gloria Padre.

A mi asesor el Dr. Jorge Hugo Calleja Gjumlich, por sus aportaciones para realizar este trabajo de investigación.

Al Dr. Carlos Aguilar Castillo por su asesoría acertada para la realización de esta tesis.

A los miembros del comité de revisión de este trabajo de tesis, el Dr. Jesús Darío Mina Antonio, Dr. Jesús Aguayo Alquicira y al Dr. Marco Antonio Oliver Salazar, gracias por sus comentarios y observaciones.

Al Dr. Ricardo Fabricio Escobar Jiménez por su asesoría relacionada con el LabView, muchas gracias por su tiempo y disposición.

A mis compañeritos de Generación: Julio Ayala, Omar Arellano, Oscar Benítez (boje), Ángel Figueroa (boje), Giovani Velasco, Amayrani Rosas, Alberto Vidal, Ricardo Schacht, Alberto Méndez y Xitla gracias por su apoyo, ayuda y por el tiempo agradable que pase con ustedes.

A los M.C: Josefa Gordillo, Susana de Leon Aldaco, Rodolfo Vargas, Alejandro Lizarraga, Juan Antonio Aquí gracias por su apoyo en sus respectivos campos y ánimo.

A mis hermanitos: Daniel Uk, Genaro Ixmatlahuac, Jose Luis Doroteo, gracias por su ánimo.

A todos mis amigos Mecánicos especialmente a Dr. Jesús Xaman, M.C. Yesenia Olazo, Royer Cundapí, Lulú Chávez, M.C. Martin Vázquez, M.C. Irving Hernández, MC. Iván Hernández, M.C. Ingrid Jiménez, M.C Ivette Zavala, M.C. Robert Alvarado muchas gracias por adoptarme, invitarme a jugar básquet, los cumpleaños y darme ánimo para concluir este trabajo.

"Al único y sabio Dios, sea gloria mediante Jesucristo para siempre. Amén".

Rom 16:27

A iglesia de Dios en México Evangelio Completo A.R (pastoreada por Noé San Martín y Maura) por sus oraciones y el privilegio de servir en sus filas, especialmente al MC. Humberto Jiménez Grajales y esposa, por ser una parte importante en mi desarrollo espiritual, personal y académico; Mayra Cantellano y Lupita Cantellano por sus oraciones, apoyo y ánimo, a Cornelio Morales por tu apoyo y animarme para este reto. Y a todos mis hermanos que formaron parte de este proceso de tesis gracias por sus oraciones.

Agradezco al Centro Nacional de Investigación y Desarrollo Tecnológico (CENIDET) por permitir dedicarme a este proyecto y proveer los recursos necesarios; a todo el personal docente, administrativo y auxiliar por brindarme los medios necesarios y facilidades para mi formación académica.

Mención especial a C. Lorena Ruiz Ramírez por agilizar los papeles, tiempo y dedicación para que se llevara a cabo este trabajo.

Al Consejo Nacional de Ciencia y Tecnología (CONACYT) que proporcionó el financiamiento para realizar este proyecto.

*“Entrad por sus puertas con acción de gracias,
Por sus atrios con alabanza;
Alabadle, bendecid su nombre.
Porque Jehová es bueno;
para siempre es su misericordia,
y su verdad por todas las generaciones”
Salmo 100:4-5*

*“Porque todo lo que es nacido
de Dios vence al mundo;
y esta es la victoria
que ha vencido al
mundo, nuestra fe”
1Jn 5:4*

ABSTRACT

This document is aimed at describing the development of a controller, based on a FPGA, for photovoltaic systems. The goal is to maximize the amount of energy harvested from the solar panels. This is accomplished by implementing the incremental conductance (INC) algorithm to track the maximum power point (MPPT), using a Spartan 3E board. The relevant aspects that were taken into account to select the tracking algorithm are described herein. The details concerning the development of the FPGA programming, and the manner in which the resources of the board were allocated are also included. It is worth point out that, for the FPGA selected, there are several Labview modules available, which greatly simplifies the programming tasks. The details of the test performed to demonstrate the validity and accuracy of the implementation, using a version of the algorithm with a fixed-length step, are at the end of this document.

RESUMEN

El presente trabajo tiene como objetivo describir el desarrollo de un controlador basado en FPGA para Sistemas Fotovoltaicos (SFV). Esto se lleva a cabo mediante la implementación del algoritmo de conductancia incremental (INC) para el seguimiento del punto de máxima potencia (MPPT) en sistemas fotovoltaicos, utilizando la tarjeta Spartan 3E. En el capítulo 1 se detalla el porqué se eligió este algoritmo para llevar a cabo esta tarea. Para la implementación se configuran los canales analógicos de manera que lleven a cabo la lectura de las señales en las entradas al FPGA. Cabe destacar que para esta tarjeta se dispone de módulos de Labview, lo que facilita la programación del algoritmo.

Se implementó satisfactoriamente el algoritmo de INC a paso fijo en el FPGA Spartan 3E. Se describen las pruebas pertinentes que demuestran el buen funcionamiento de éste, variando el voltaje de entrada al FPGA.

Contenido

ABSTRACT	i
RESUMEN	ii
Capítulo 1	8
Introducción	8
1.1 Módulo Fotovoltaico y MPP	10
1.2 Planteamiento del problema.....	14
1.3 Estado del Arte.....	15
1.4 Propuesta de solución	26
1.5 Objetivo general.....	27
1.6 Justificación	27
1.7 Referencias.....	28
Capítulo 2	30
Implementación para el algoritmo INC.....	30
2.1 Detalle del algoritmo	30
2.1.1 Algoritmo conductancia incremental (INC) con paso fijo	30
2.1.2 Funcionamiento del algoritmo	34
2.1.3 Sistema para el control de MPP.	39
2.1.3.1 El funcionamiento del Push-Pull.	39
2.2 Detalle del hardware	41
2.2.1 FPGA Spartan 3E.....	41
2.2.1.1 Opciones de configuración del FPGA	43
2.2.1.2 Expansión de conectores.....	44
2.2.1.3 6 pines de accesorio de salida (Accessory Headers).....	45

2.2.1.4 Módulo ADC (Convertidor analógico-digital) en Spartan 3E FPGA.....	45
2.2.2 Programación gráfica Labview.....	53
2.2.2.1 Estructuras de Labview.....	54
2.3 Bibliografía	55
Capítulo 3	56
Programación en el FPGA	56
3.1 Limitaciones en la paleta del VI del FPGA.	56
3.1.1 Modo PC interactiva (Host)	57
3.2 Programación del FPGA	58
3.2.1 Configuración del Preamplificador FPGA.....	59
3.2.2 Programa para el ADC	62
3.2.3 Programación del algoritmo INC	64
3.2.4 Programa Host para algoritmo INC.....	67
3.3 Bibliografía:	70
Capítulo 4	71
Resultados	71
4.1 Caso 1: izquierda del MPP	72
4.2 Caso 2: derecha MPP.....	74
4.3 Caso 3: disminución de irradiancia.....	78
4.4 Caso 4: aumento de irradiancia.....	80
Capítulo 5	84
Conclusiones.....	84
Anexo A: Convertidor Push-Pull	87
Anexo B:Memoria utilizada en FPGA	90

Lista de figuras

Figura 1.1 Circuito equivalente de una celda solar.....	10
Figura 1.2 La curva característica del panel solar.....	13
Figura 1.3 Curvas I-V variando las condiciones climáticas.....	13
Figura 1.4 Curvas V-P variando las condiciones climáticas.....	14
Figura 1.5 Sistema de MPPT.....	21
Figura 1.6 Formas de onda de tensión, corriente y potencia con tamaño de paso variable.....	22
Figura 2.1 Diagrama de flujo del método INC.....	33
Figura 2.2 Caso 1: izquierda del MPP.....	35
Figura 2.3 Caso 2: derecha del MPP.....	36
Figura 2.4 Caso 3: disminución de irradiancia.....	37
Figura 2.5 Caso 4: aumento de irradiancia.....	38
Figura 2.6 Sistema general para la implementación del FPGA.....	39
Figura.2.7 Convertidor Push-Pull.....	40
Figura 2.8 Formas de Onda del convertidor Push-Pull.....	42
Figura 2.9 FPGA Spartan 3E.....	43
Figura 2.10 Opciones de configuración detallada.....	44
Figura 2.11 Conectores de expansión.....	44
Figura 2.13 Canales de captura del circuito analógico.....	46
Figura 2.14 Comunicación via SPI.....	47
Figura 2.15 Circuito de captura de señales análogas.....	48
Figura 2.16 Interface serial SPI para el amplificador.....	50
Figura 2.17 Diagrama de tiempos para la comunicación con el amplificador.....	50
Figura 2.18 Representación de la salida del ADC.....	52
Figura 2.19 Interface de conversión analógico a digital.....	53
Figura 2.20 Estructura flat sequence.....	54
Figura 3.1 Interacción FPGA y Host.....	57
Figura 3.2 Host.....	57
Figura 3.3 Diagrama a bloques para adquisición y almacenamiento del voltaje analógico.....	59
Figura 3.6. Contador hasta 8 para terminar de configurar el preamplificador.....	60
Figura 3.7 Ganancia en LEDs.....	61
Figura.3.8 SubVI preamplificador.....	61
Figura 3.9 Adquisición voltaje analógica (parte I).....	62
Figura 3.10 Adquisición voltaje analógica (parte II).....	63
Figura 3.11 Adquisición voltaje analógico (parte III).....	63
Figura 3.12 SubVI ADC.....	63
Figura 3.13 Programa INC para Adquisición y almacenamiento de voltajes y corrientes.....	64
Figura 3.14 Señales de los MOSFET del Push-Pull.....	65
Figura 3.15 Programa del PWM.....	66
Figura 3.16 Programa de actualización del PWM.....	66
Figura 3.17 Memoria D.....	67
Figura 3.18 Diagrama de bloques del programa Host.....	68
Figura 3.19 Programa Host del algoritmo INC.....	69
Figura 3.20 Panel frontal del Host.....	69
Figura 4.1 Panel frontal D=0.3.....	72
Figura 4.2 Señal de osciloscopio a D=0.3.....	72
Figura 4.3 Disminuye D, Dout=D-step=0.3-0.05=0.25.....	73
Figura 4.4 D=0.25 actualizado.....	74
Figura 4.5 D=0.25, Dout= Data.....	74
Figura 4.6 D=0.24 actual en panel frontal.....	75

Figura 4.7 $D=0.24$ actual en osciloscopio	75
Figura 4.8 $D_{out}=D_{actual}+step$ (0.05).....	76
Figura 4.9 $D=0.29$ actualizado en panel frontal.....	77
Figura 4.10 $D=0.29$ actualizado en osciloscopio.....	77
Figura 4.11 $D=0.20$ en panel frontal.....	78
Figura 4.12 $D = 0.20$ en osciloscopio.....	79
Figura 4.13 $D_{out}= D-step = 0.15$	79
Figura 4.14 $D=0.15$ Actualizado	80
Figura 4.15 $D_{data}=D_{actual}$ Actualizado	80
Figura 4.16 $D=0.20$, caso 4 en panel frontal.....	81
Figura 4.17. $D = 0.20$, caso 4 en osciloscopio.....	81
Figura 4.18 $D_{out}= D+0.05=0.25$	82
Figura 4.19. $D = 0.25$ actualizado en panel frontal.....	82
Figura 4.20. $D = 0.25$ actualizado en osciloscopio.....	83
Figura A.1 Convertidor push-pull.....	87
Figura A.2 Tiempos de conmutación de Q1y Q2	87
Figura A.3 Circuito equivalente 1 encendido	88
Figura A.4 Circuito equivalente 2 y 4, tiempo muerto.....	88
Figura A.5 Circuito equivalente 3.....	88
Figura B.1 Memoria utilizada del FPGA.....	90

Lista de tablas

Tabla 1.1 Características de las diferentes técnicas de MPPT.....	17
Tabla 2.2 Ganancias del amplificador.	49
Tabla 2.3. Líneas para deshabilitar otros dispositivos en el bus SPI.	53

Introducción

En la actualidad ha cobrado relevancia la preservación del medio ambiente. El interés surge debido a la alta contaminación que generan los automóviles y las empresas al fabricar sus productos. Como una respuesta tecnológicamente sensata las energías renovables han crecido durante los últimos años. Hay diferentes tecnologías que no perjudican al medio ambiente, pero las que han alcanzado la madurez suficiente para estar en el mercado con un grado de competitividad aceptable son la energía fotovoltaica y la eólica.

La implementación de energía eólica es más económica que la fotovoltaica. En el caso específico de México, la desventaja radica en que no en todos los lugares corre el viento necesario para que la inversión sea redituable. En cambio, la energía fotovoltaica tiene la ventaja de que en cualquier parte de nuestro país se encuentra este recurso solar, resaltando que en los estados del norte es donde se obtiene mayor irradiancia.

Así pues, hablar de sistemas fotovoltaicos (SFV) es un tema relevante en la actualidad y el sector empresarial está invirtiendo en esta tecnología. A corto plazo y en contraste con los demás elementos involucrados, la inversión más alta de un SFV se asocia con la compra de los paneles solares, aunque tienen la ventaja de ofrecer vidas útiles superiores a los veinte años. Se considera entonces que a largo plazo esta tecnología es redituable, sobre todo para usuarios que tienen altos consumos de energía eléctrica.

La energía eléctrica generada por los SFV es directamente proporcional a la irradiancia que incide sobre el panel solar. La irradiancia depende a su vez de las condiciones atmosféricas y de la posición del sol en el horizonte: en el transcurso del día, sólo en algunos lapsos se aproximará al valor máximo posible, cercano al nominal de los paneles. Esto ocurrirá cuando el sol esté en el cenit (alrededor del medio día) y considerando un día soleado. En cambio, si el día está parcialmente nublado, habrá variaciones en la irradiancia y disminuirá la potencia en los paneles. A lo largo del día, para cada combinación de irradiancia y temperatura existirá un valor de potencia máxima que el panel es capaz de entregar. Este valor es el punto de máxima potencia MPP (por sus siglas en inglés); para obtener la mayor energía posible del SFV se emplea un algoritmo que asegure la operación constante en el MPP. El dispositivo que se utilizará para esta tarea será la tarjeta FPGA (Field Programmable Gate Array) la cual, aunque existe desde hace algunos años, está actualmente en auge en el mercado. A diferencia de otras tarjetas programables, las ventajas del FPGA son las siguientes: facilidad de reconfiguración, expansibilidad, sin límite en cuanto al número de convertidores analógico-digital ni contadores

que soporta, etc. Estas características serán de utilidad para alcanzar con rapidez el MPP ante cambios en la irradiancia, lo que redundará en un mejor aprovechamiento de la energía.

1.1 Módulo Fotovoltaico y MPP

La unidad estructural básica del módulo solar son las celdas fotovoltaicas. Una celda solar convierte la energía de los fotones de la luz solar en electricidad por medio del fenómeno fotoeléctrico, el que se presenta en algunos materiales semiconductores, como el silicio y el selenio.

Una sola celda solar puede producir solamente una pequeña cantidad de potencia. Para incrementar la potencia de salida de un sistema, las celdas solares se conectan generalmente en combinaciones en serie y en paralelo para formar módulos fotovoltaicos [1].

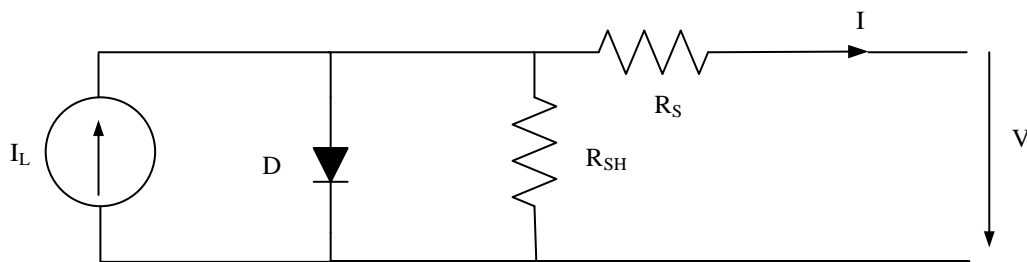


Figura 1.1 Circuito equivalente de una celda solar.

La celda solar se puede representar por medio del modelo eléctrico de la figura 1.1. Su característica corriente-voltaje se expresa por la siguiente ecuación:

$$I = I_L - I_0 \left(e^{\frac{q(V-IR_s)}{AkT}} - 1 \right) - \frac{V - IR_s}{R_{SH}} \quad (1.1)$$

donde I y V son la corriente y la tensión de salida respectivamente de la celda solar, I_0 es la corriente de saturación en sombra, q es la carga de un electrón, A es el factor de la calidad del diodo (ideal), k es la constante de Boltzmann, T es la temperatura absoluta, I_L es la corriente generada por la luz que incide sobre ésta, R_S y R_{SH} son las resistencias serie y paralelo de la celda solar.

El término R_S corresponde a la resistencia que presentan los contactos y el material semiconductor de la celda solar. El origen de la resistencia paralelo R_{SH} es más difícil de explicar. Se relaciona con el carácter no ideal de la unión p-n y la presencia de impurezas cerca de los bordes de la celda, las que proporcionan una trayectoria de cortocircuito alrededor de la unión. En un caso ideal R_S sería cero y R_{SH} infinito; desafortunadamente, este escenario ideal no es posible y los fabricantes tratan de minimizar el efecto de ambas resistencias para mejorar sus productos.

A veces, para simplificar el modelo, el efecto de la resistencia en paralelo se minimiza; es decir, R_{SH} se considera infinita, por lo que el último término de (1.1) desaparece.

Teniendo en cuenta esta simplificación, la característica corriente-voltaje de un panel FV se expresa por la ecuación (1.2), donde n_p y n_s son el número de celdas solares en paralelo y serie, respectivamente.

$$I \approx n_p I_L - n_p I_0 \left(e^{\frac{q(v-IR_s)}{AkTn_s}} - 1 \right) \quad (1.2)$$

Deben señalarse dos puntos importantes de la característica corriente-tensión: el voltaje de circuito abierto V_{OC} y la corriente de cortocircuito I_{SC} . En ambos puntos la potencia generada es cero. V_{OC} se puede aproximar a partir de (1.1), considerando que la corriente de salida de la celda es cero (es decir: $I = 0$) y despreciando la resistencia en paralelo R_{SH} ; el resultado es la ecuación (1.3). La corriente de cortocircuito I_{SC} es la corriente en $V = 0$ y es aproximadamente igual a la corriente generada por luz I_L como se muestra en la ecuación (1.4).

$$V_{oc} \approx \frac{AkT}{q} \ln \left(\frac{I_L}{I_0} + 1 \right) \quad (1.3)$$

$$I_{sc} \approx I_L \quad (1.4)$$

La potencia máxima es generada por la celda solar en un punto de voltaje-corriente, donde el producto V-I es máximo. Este punto se conoce como el MPP (punto máximo de potencia) y es único, como puede verse en la figura 1.2 (donde están representados también V_{OC} e I_{SC}). Dos factores importantes que deben tenerse en cuenta son la irradiación y la temperatura debido a que afectan fuertemente las características de los módulos solares, (figura 1.3). La figura 1.4 muestra como afecta a la potencia el cambio de irradiancia y temperatura. Como resultado, el MPP varía durante el día y es la razón principal por la cual el MPP debe seguirse constantemente, asegurando así que se obtenga la máxima potencia disponible del panel [2].

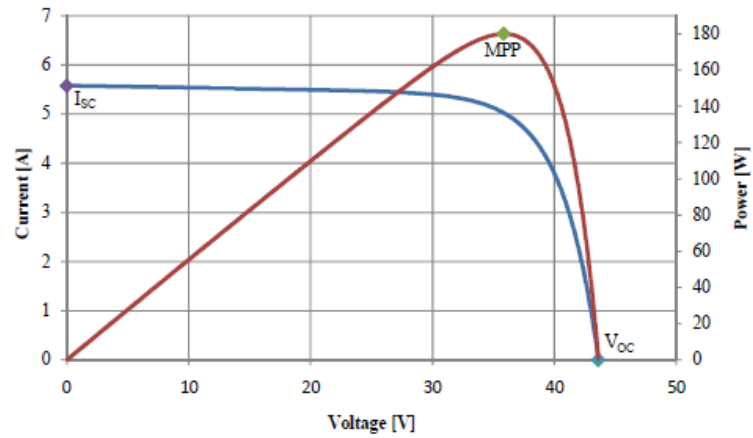


Figura 1.2 La curva característica del panel solar

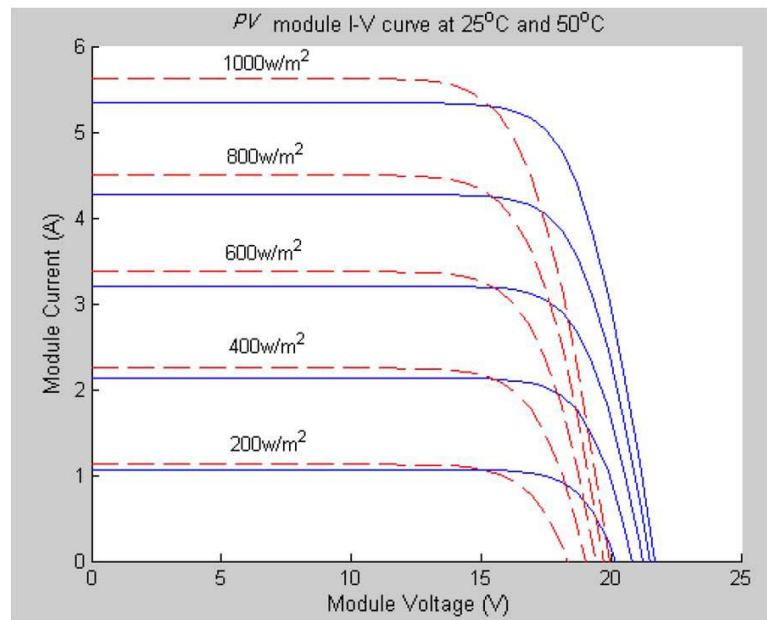


Figura 1.3 Curvas I-V variando las condiciones climáticas

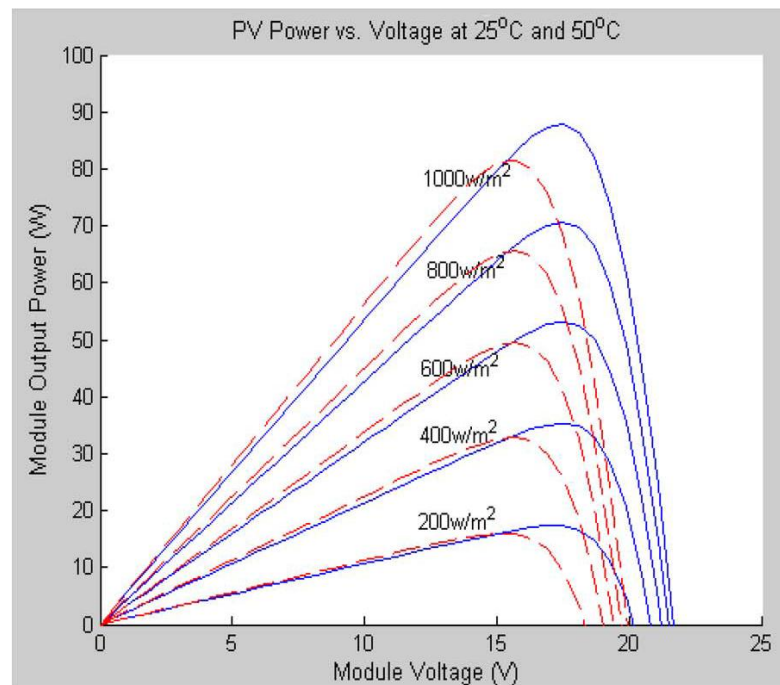


Figura 1.4 Curvas V-P variando las condiciones climáticas

1.2 Planteamiento del problema

En un sistema fotovoltaico los elementos más costosos son los paneles solares, cuya eficiencia oscila entre un 12 % y un 15 %, dependiendo de la tecnología. Para hacer más atractiva la energía solar-eléctrica se requiere acelerar la recuperación de la inversión y alcanzar la paridad con la energía eléctrica generada por medios convencionales. Además como se mostró anteriormente la potencia obtenida del panel solar se ve afectada fuertemente ante la variación de irradiancia y de la temperatura. Estas condiciones varían constantemente a lo largo del día por lo que no se obtendrá la mayor potencia posible del panel FV. Para alcanzar esta meta se utiliza un algoritmo seguidor del Punto de Potencia Máxima.

El algoritmo debe cumplir con los siguientes requerimientos:

- Realizar la medición real del MPP
- Sin oscilaciones en régimen permanente
- Con buen comportamiento dinámico ante variaciones súbitas de irradiancia
- Mínima cantidad de sensores
- Un nivel de complejidad accesible al usuario

Pueden seguirse diferentes métodos para alcanzar el MPP y en la literatura técnica se han reportado entre 10 y 12 de ellos. Dado el abanico de posibilidades, es necesario evaluar las propuestas reportadas y elegir el más adecuado para la aplicación. Se incorporarán las modificaciones pertinentes y se implementará el algoritmo en una tarjeta FPGA comercial.

1.3 Estado del Arte.

A continuación se describen los artículos reportados en la literatura, relacionados con los algoritmos que se utilizan para alcanzar el MPP. En la primera fase del análisis se investigaron los algoritmos propiamente dichos, y posteriormente las implementaciones de éstos en FPGA.

En [3] se describen 5 algoritmos para seguir el MPP, los cuales se mencionan brevemente.

- *Algoritmo perturbar y observar (P&O)*. Es de implementación sencilla, pero presenta dificultades en la medición del MPP ante variaciones repentinas de la irradiancia solar. Además, también presenta oscilaciones cuando alcanza el MPP, lo que se refleja en pérdidas de potencia. Se reportan algunas mejoras para disminuir las oscilaciones en régimen permanente, pero la respuesta ante los cambios de irradiancia continua siendo relativamente lenta y menos eficiente para alcanzar el MPP.

- *Corriente y voltaje constante.* Se propone establecer una relación entre el voltaje del punto de máxima potencia V_{MPP} y el de circuito abierto V_{OC} , asignando a éste una constante k , la cual varía ante los cambios de irradiancia. Es de implementación sencilla (no es necesario medir la corriente que entrega el panel), pero la eficiencia para el seguimiento de MPP es baja comparada con otros algoritmos.
- *Celda piloto.* Usa el método de corriente y voltaje constantes. Las mediciones de corriente de corto circuito I_{SC} y V_{OC} se llevan a cabo en una celda piloto cuyas características son similares a las de las celdas en un arreglo solar de mayor tamaño. Sin embargo, el problema del valor de k se presenta todavía. Además, este método tiene un inconveniente logístico, debido a que los parámetros de la celda piloto deben aparearse cuidadosamente con los del arreglo FV; la celda piloto debe calibrarse, lo que aumenta el costo del sistema.
- *Conductancia incremental.* El algoritmo de la conductancia incremental se obtiene derivando la potencia del arreglo FV con respecto a la tensión e igualando el resultado a cero. La ventaja con respecto al algoritmo de P&O es que determina la dirección en la que debe aplicarse la corrección para desplazar el punto de operación hacia el MPP. Una vez que alcanza el MPP, el algoritmo continúa operando en este punto hasta que se detecte un cambio en la corriente. Además, cuando las condiciones atmosféricas cambian rápidamente, no realiza un seguimiento en dirección equivocada, por lo cual no oscila una vez que llega al MPP.
- *Capacitancia parásita:* El algoritmo de la capacitancia parásita CP es similar al de conductancia incremental, excepto por el efecto de la capacitancia parásita en la unión de la celda solar; se incluyen los modelos de almacenamiento de carga en la unión p-n de las celdas solares.

En [2] se presentan las características de 12 algoritmos para alcanzar el MPP: perturbar y observar, conductancia incremental, por medio de lógica difusa, con redes neuronales, voltaje fraccional de circuito abierto, corriente fraccional de corto circuito, barrido de corriente, Punto de Potencia Máxima de corriente y voltaje computacional, técnica de MPPT basado en estados, y búsqueda de múltiple máximo. Se concluye que es preferible utilizar los más comunes: P&O, INC y lógica difusa.

De igual manera, en [4] se presentan 6 algoritmos para alcanzar el MPP: P&O, INC, voltaje fraccional de circuito abierto, corriente fraccional de corto circuito, control de lógica difusa, red neuronal. En la tabla 1.1 se compara los algoritmos mencionados, mostrándose sus características principales.

Tabla 1.1 Características de las diferentes técnicas de MPPT.

Técnica de MPPT	Velocidad de convergencia	Complejidad en la implementación	Sintonización periódica	Parámetros sensados
Perturbar y observar	Variable	Baja	No	Voltaje
Conductancia Incremental	Variable	Media	No	Voltaje, corriente
V_{oc} fraccional	Media	Baja	Sí	Voltaje
I_{sc} fraccional	Media	Media	Sí	Corriente
Control de lógica difusa	Rápida	Alta	Sí	Varios
Redes neuronales	Rápida	Alta	Sí	Varios

En [5] se comparan los algoritmos de P&O e INC aplicados a los convertidores *Buck* y *Boost*. Los resultados demostraron que el algoritmo INC tiene mejor funcionamiento que el P&O. Con el convertidor *Boost* el algoritmo INC muestra una salida más estable; sin embargo, el algoritmo de P&O obtiene una salida de valor más alto que el de INC. En general, el INC presenta mejores resultados en comparación del P&O.

Para aplicaciones con almacenamiento de energía, en [6] se justifica el hecho de utilizar controladores de MPPT para evitar daños a las baterías, debido a los picos de voltaje. Una forma para obtener mayor potencia de un SFV podría ser aumentar el número de paneles FV, lo que sería costoso y de ahí la importancia de estas técnicas. Se muestran las características de los algoritmos P&O e INC, los diagramas de flujo, la programación empleada, su implementación y los circuitos empleados para los sensores.

En [7] se presenta una mejora del algoritmo INC, identificada como dP-INC, la cual se usa en un controlador de voltaje para un inversor FV. Este algoritmo puede detectar si la corriente demandada al panel FV se ubica en la cercanía del MPP o no, y ajusta el incremento en la referencia de voltaje. Este método sigue el MPP con una precisión de hasta el 99.9% en el caso de alta irradiancia, aunque la eficiencia baja a 99.6% cuando la irradiancia es de 50 W/m^2 . También se comprueba un mejor desempeño al seguir el MPP en comparación al algoritmo de P&O.

Además del algoritmo INC, en [8] se desarrolló recientemente un método semejante llamado MPPT por incremento de resistencia (INR). Este algoritmo trabaja también con la variación de potencia ΔP , pero utiliza la variación de corriente ΔI , en lugar de la variación de voltaje ΔV utilizada en el algoritmo de INC. Se puede decir que entre ambos métodos existe una relación de dualidad. El algoritmo de INR plantea las ecuaciones para calcular el ciclo de trabajo D , el cual se modifica en un convertidor *Boost*. El sistema se simuló en *Simulink* utilizando el convertidor *Boost* para la etapa de potencia y un panel de 110W. Se reportan pruebas del INR con pasos tanto de tamaño fijo como variable. Se demuestra que con paso de tamaño variable se obtienen menores pérdidas de potencia y menor oscilación cuando se alcanza el MPP, en comparación con el paso fijo. También se prueba que al aumentar el factor de escalamiento N se obtienen mejores

resultados. Se reportan pruebas de laboratorio en el SFV mencionado en el artículo, con un arreglo de 3 paneles en paralelo. Se muestran resultados favorables al probarse el sistema bajo variaciones de potencia abruptas. De igual manera, se comprueba el funcionamiento dinámico del algoritmo con pasos de tamaños variable y fijo, obteniéndose las mismas conclusiones que en la simulación. El sistema de control utilizado es un microcontrolador C515C y la frecuencia de conmutación es 50 kHz.

Para mejorar el comportamiento del algoritmo INC, en [9] se muestra un nuevo algoritmo denominado incremento de potencia ayudado por conductancia incremental, identificado como PI-INC. El algoritmo funciona utilizando una frecuencia variable con D constante (*Variable-Frequency Constant Duty Control*, VFCD) o una frecuencia constante con D variable (*Constant-Frequency Variable-Duty Control* CFVD). En este sistema se utilizan dos convertidores *flyback* por medio de un sistema de control del dsPIC33FJ06GS202, utilizando dos paneles de 250 W.

En [10] se reporta un algoritmo con dos lazos de control independientes para el control de MPP. El primer lazo contiene el algoritmo de MPP y el segundo es un control proporcional, o proporcional integral. Como se mencionó, el algoritmo de conductancia incremental se obtiene derivando la potencia del arreglo FV con respecto a la tensión e igualando el resultado a cero. Debido a que la mayoría de los puntos de operación no corresponde a este valor, el principal propósito del segundo lazo de control es hacer este valor cercano a cero. Este lazo de control tiene las siguientes características: operación simple, facilidad de diseño, facilidad de mantenimiento entre otras, y es muy utilizado para sistemas lineales.

El SFV es un sistema no lineal debido a que su salida cambia dependiendo de las condiciones atmosféricas, las cuales no son lineales, por lo que este control no trabaja de manera adecuada.

Por esta razón se propone eliminar el control, proponiendo para ello un margen pequeño de error en lugar de cero y obtener mejores resultados.

En este artículo se considera utilizar un convertidor cuk o un buck-boost para la etapa de potencia; sin embargo se eligió el cuk por tener bajas pérdidas en la conmutación y mejor eficiencia entre los convertidores no aislados. El algoritmo propuesto se simula en *Simulink* para comprobar la eficiencia del SFV bajo variaciones de irradiancia y se compara con los resultados experimentales.

Las pruebas se hicieron conectando el sistema a un arreglo de cuatro paneles en serie, se incrementó y decrementó el número de paneles para observar el comportamiento de la corriente, voltaje y potencia. Cabe destacar que el algoritmo empleado es INC de tamaño de paso fijo y se elimina el segundo lazo de control. Para el control se utilizó el DSP TMS320F2812.

Generalmente el tamaño del paso es fijo para el algoritmo de INC. Se sabe que el empleo de un paso de tamaño grande contribuye a una dinámica rápida, pero presenta el inconveniente de oscilar excesivamente en régimen permanente, lo que disminuye su eficiencia. Para resolver este problema se emplea el tamaño de paso variable, como se presenta en [11]. En este artículo se simuló el algoritmo de INC de paso variable en *Simulink* para verificar su comportamiento. Se compara el INC con paso fijo ante cambios de irradiancia, y se observa claramente cómo oscila la potencia cuando se alcanza el MPP. De igual manera, se simula el algoritmo INC de paso variable modificando el valor N , con lo cual se obtienen mejores resultados. Además, se grafican los cambios en el ciclo de trabajo con este tamaño de paso. Se construyó el sistema de la figura 1.5 con tres módulos en serie y un convertidor *push-pull* (se proporcionan los datos de diseño), al cual el algoritmo INC modifica el ciclo de trabajo. El sistema de control se implementa mediante la tarjeta DSP TMS320LF2407.

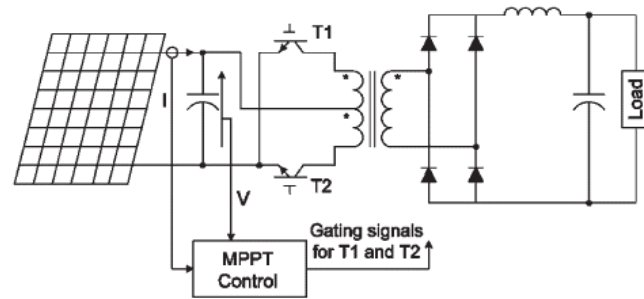


Figura 1.5 Sistema de MPPT

En la figura 1.6 se muestra el incremento de corriente y potencia con pasos grandes; esto se debe al tamaño del paso cuando inicia el algoritmo. Este comportamiento se puede mejorar agregando una constante de seguimiento de voltaje *CVT* (*Constant Voltage Tracking*), como se muestra en la figura 1.7. Este algoritmo suaviza visiblemente las señales dado que inicia al 80% de la señal, como se muestra en la figura 1.8.

Para finalizar, se comprueba experimentalmente la eficiencia del método de MPPT con tamaño de paso variable y fijo, pasando de dos a tres módulos FV y después de 3 a 1 módulo. Se obtienen mejores resultados con el algoritmo de paso variable.

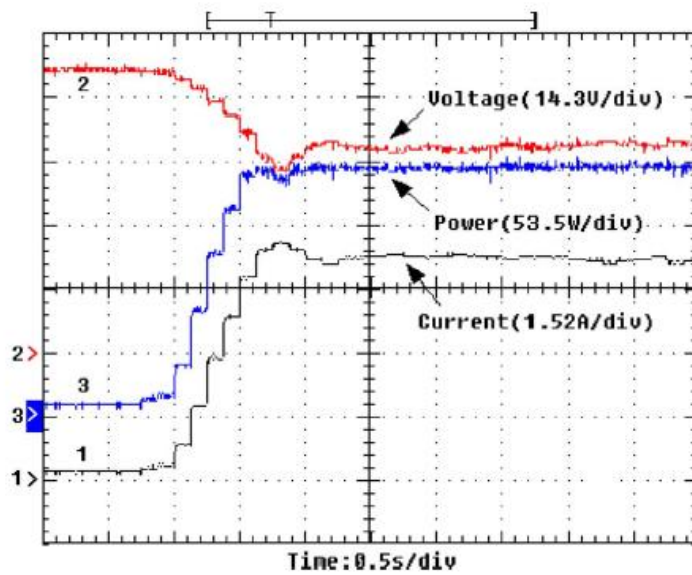


Figura 1.6 Formas de onda de tensión, corriente y potencia con tamaño de paso variable

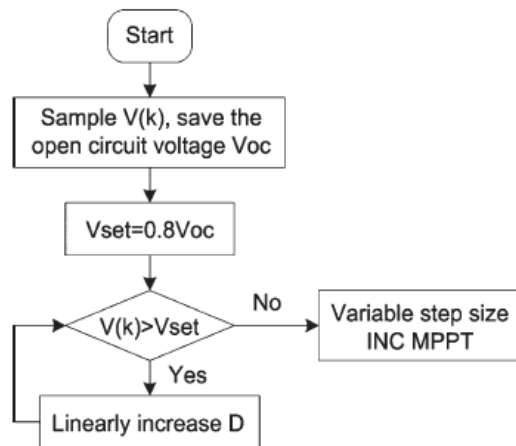


Figura 1.7 Programa CVT

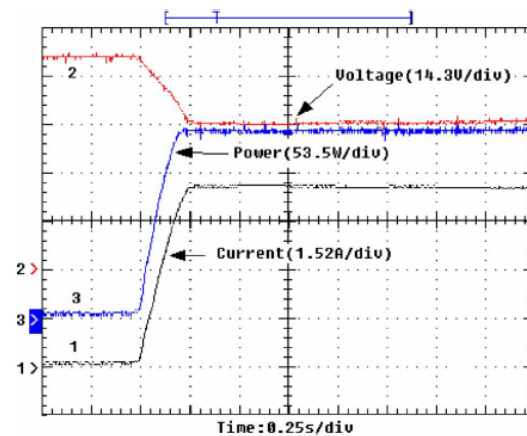


Figura 1.8.-Inicio formas de onda con el algoritmo de MPPT con paso tamaño variable y CVT

En [12] se implementa el algoritmo P&O, en una tarjeta FPGA Virtex-II-Pro (xc2v1000-4fg456), la cual se programó mediante el lenguaje VHDL. Se utilizó un convertidor *boost* como etapa de potencia, en el cual se varía el ciclo de trabajo. En lugar de utilizar sensores comerciales,

se emplean circuitos sencillos para sensar la corriente y el voltaje, lo que disminuye el costo del sistema. Para la conversión analógico-digital se utilizan dos ADC0804 de 8 bits. Se muestran gráficas de potencia donde se compara el algoritmo ante diferentes niveles de irradiancia. Posteriormente se realizan pruebas variando el ciclo de trabajo de 0.4 y 0.5, obteniéndose eficiencias de 96.13% y 95.23%, respectivamente. Se concluye que hay que mantener un compromiso entre la velocidad para alcanzar el MPP y la eficiencia del sistema cuando hay cambios rápidos de irradiancia.

En [13] se emplea el algoritmo de lógica difusa y se implementa en el FPGA Xilinx Virtex-II (XC2v1000-4fg456) en la tarjeta Virtex-IITMV2MB1000. Esta tarjeta contiene 10,240 celdas lógicas y 40 multiplicadores de 18 por 18. Se realizan simulaciones comparando los algoritmos de lógica difusa y P&O, obteniéndose mejores resultados del primero. Sólo se implementa el algoritmo en el FPGA y no se comprueba el comportamiento de manera experimental.

En [14] se demuestra que el algoritmo INC mejora variando el ciclo de trabajo en vez del voltaje. Se describe el INC empleado en un SFV de bombeo aislado, con un arreglo FV de 1080 W de potencia pico conectado a un motor de CD de imanes permanentes acoplado a una bomba centrífuga. Se enfoca particularmente a proporcionar la evaluación de dos métodos comúnmente utilizados: perturbación del voltaje de referencia y perturbación del ciclo de trabajo.

Se muestra la comparación del INC con perturbación de voltaje o del ciclo de trabajo D , alcanzándose mejor eficiencia con la variación de D . Además, utiliza un controlador PI, el cual evita oscilaciones y contribuye a mejorar la eficiencia.

En [15] se presenta el algoritmo de INC se basa en la perturbación del voltaje. Se realizan las pruebas con el algoritmo de “*Hill climbing*” e INC usando 2 enfoques diferentes, uno probando

con un simulador FV y el otro usando dos sistemas FV idénticos. Ambos algoritmos se diseñaron para las mismas condiciones atmosféricas reales, con un alto nivel de fluctuación de la irradiancia (aproximadamente 30 W). El algoritmo “*hill climbing*” fue ligeramente superior al INC. Ambos algoritmos se implementaron con inversores conectados a la red y se probaron sin encontrar una diferencia significativa en el campo. En conclusión, no se muestra alguna superioridad significativa del algoritmo de “*hill climbing*” sobre el de INC.

En [16] se presenta la implementación de un SFV de una sola etapa conectado a la red trifásica, con un método modificado de MPPT y compensación de potencia reactiva. Se realiza la compensación de la potencia reactiva de una carga local para mitigar la carga de la red. El algoritmo de INC muestrea el voltaje y la corriente del panel FV para calcular la potencia de salida P_{PV} . Se compara $P_O - P_{PV} > \Delta p$, para incrementar o decrementar el tamaño del paso P_{REF} (potencia de salida del inversor). Δp es el límite del cambio de paso, P_o es la potencia de salida en el previo periodo de control. Se describen pruebas en un SFVI real con una unidad de compensación de potencia reactiva, con el algoritmo de INC de paso fijo y paso variable. Se obtienen gráficas del voltaje del capacitor de enlace mejores con paso variable, manteniéndose un voltaje más o menos constante en el capacitor. En términos generales el INC es el mismo, se muestran ecuaciones y se verifica el punto de operación del MPP con paso variable.

En [17] Se presenta el factor de potencia de la salida de un inversor fotovoltaico, el cual se conecta a la red eléctrica y se controla mediante la técnica de control con linealización de la retroalimentación (Feedback Linearization Control, FLC). Esta técnica transforma el modelo del estado no lineal del inversor referenciado en dos subsistemas lineales equivalentes; se ubica un polo en el lazo de control del subsistema para controlar separadamente el control del factor de

potencia y el voltaje de enlace de CD del inversor. El MPP se basa en el algoritmo INC de paso variable. Se muestra el algoritmo normal del paso variable del INC perturbando el voltaje de referencia. El control es externo al MPP, se muestran graficas donde se comparan tanto el paso fijo como el variable resultando mejor el variable para el algoritmo de INC. Se demuestra que la THD de la corriente de la red es menor a la que se obtiene con el algoritmo de paso fijo. Se presenta sólo simulación y se propone a futuro implementar el algoritmo con un FPGA.

En [18] se presenta el algoritmo INC, proponiéndose un control PI con series de Taylor. A partir de simulaciones en PSCAD/EMTDC se deduce el algoritmo con una matriz de estados. Se muestra la eficiencia del INC y la manera en que es altamente robusto. Lo relevante es la originalidad del método de linealización, ya que deja los términos de segundo orden. A primera vista, se aprecia algo complejo para llevarlo a cabo.

Conclusión de la literatura.

En la búsqueda de algoritmos para alcanzar el MPP en SFV se encontraron 12 técnicas [2-7]. Las más populares por el número de publicaciones son: perturbar y observar (P&O), conductancia incremental (INC) y lógica difusa. Esto se debe a que estos algoritmos son simples y miden el MPP en tiempo real. De estos tres, el algoritmo de lógica difusa es el más complicado y el diseñador necesita experiencia para implementarlo; el algoritmo de P&O es el más fácil de implementar, pero presenta pérdidas de potencia cuando existen cambios súbitos de irradiancia.

En contraste, para el algoritmo de INC la implementación es de complejidad media, se determina la dirección en que debe producirse la corrección en el punto de operación que permita alcanzar el MPP, por lo cual presenta buenos resultados para responder a cambios repentinos de

irradiancia. Por estas razones en este trabajo de tesis se optó por usar el algoritmo de INC e implementarlo en el FPGA.

Se revisaron además los artículos [8-13] donde se describen mejoras al algoritmo INC de MPPT. En [10] el algoritmo de PI-INC presenta mejor seguimiento para alcanzar el MPP comparado con el algoritmo de INC, por lo cual ofrece menos pérdidas a la salida. Sin embargo, tiene la desventaja de utilizar dos convertidores CD-CD, siendo más costoso y además se modifica la frecuencia en una de las señales de control. Ésto no es recomendable para este estudio, debido a que las señales de control el convertidor push-pull son PWM, por lo cual la mejor opción presentada es el algoritmo INC con paso de tamaño variable [12]. Se puede notar en los artículos mencionados que estos algoritmos se implementan sólo en microcontroladores y en DSP. Aunque en [14 y 15] se muestra los algoritmos de MPPT más populares (P&O y de lógica difusa) no se encontró alguno con el algoritmo de INC con paso de tamaño variable implementado mediante un FPGA.

En los artículos [16-18] se desarrolla una búsqueda de alguna modificación del algoritmo de conductancia incremental (INC) para obtener una mayor potencia del proceso, además se muestran algunas aplicaciones con el algoritmo (INC), haciéndose también comparaciones con algún otro algoritmo.

1.4 Propuesta de solución

La implementación del algoritmo de INC es de complejidad media; además, se determina la dirección en que debe producirse la corrección que permita alcanzar el MPP, lo cual favorece a obtener una respuesta rápida ante cambios repentinos de irradiancia. Por estas razones, es éste el que se implementa en el presente trabajo de tesis.

1.5 Objetivo general

Implementar en un FPGA el algoritmo de conductancia incremental (INC) de seguimiento del punto de máxima potencia para sistemas fotovoltaicos.

1.6 Justificación

Las líneas de investigación en CENIDET se centran en el estudio de técnicas y alternativas para el “Ahorro y Uso Eficiente de la Energía Eléctrica” en fuentes convencionales y alternas de energía. En fuentes no convencionales de energía se trabaja en Sistemas Fotovoltaicos (SFV) y Eólicos [20].

Como se mencionó anteriormente, los SFV tienen poca eficiencia (alrededor del 12%) y las características de los paneles solares se ven afectadas por dos factores: la irradiancia y la temperatura que incide sobre él, por lo cual, se necesitan métodos para aprovechar el máximo punto de potencia durante el transcurso del día. En la literatura se estudiaron 13 algoritmos para llevar a cabo esta tarea; las más populares por el número de publicaciones y aplicaciones son: perturbar y observar P&O, conductancia incremental y lógica difusa. Se decidió implementar el algoritmo de INC debido a que no presenta oscilaciones en régimen permanente; presenta una complejidad media con respecto al de lógica difusa, y no se necesita tener experiencia en el desarrollo de éste. Además, se planea implementar el INC en la plataforma de Labview debido a que cuenta con módulos y drivers para el Spartan 3E, lo que facilita la programación.

1.7 Referencias

- [1] Safari and S. Mekhilef, "Simulation and Hardware Implementation of Incremental Conductance MPPT With Direct Control Method Using Cuk Converter," *Industrial Electronics, IEEE Transactions on*, vol. 58, pp. 1154-1161, 2011.
- [2] D.S.Morales, "Maximum Power Point Tracking Algorithms for Photovoltaic Applications", Thesis submitted for examination for the degree of Master of Science in Technology, pp 22-35, 2010.
- [3] D. P. Hohm and M. E. Ropp, "Comparative Study of Maximum Power Point Tracking Algorithms", progress in photovoltaics: research and applications, pp. 47-62, 2003.
- [4] A.Harjai, A.Bhardwaj, M.Sandhibig "study of maximum power point tracking (mppt) techniques in a solar photovoltaic array, Department of Electrical Engineering National Institute of Technology Rourkela, pp22-33, 2011.
- [5] A.K. Yadav, S.Thirumaliah, G.Haritha, "Comparison of MPPT Algorithms for DC-DC Converters Based PV Systems" ,*International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering* ,Vol. 1, pp.18-23, 2012.
- [6] M. R.Hamzescu, S. Oprea, "Practical Guide to Implementing Solar Panel MPPT Algorithms", *Microchip Technology Inc*, pp1-16, 2013.
- [7] T. Kerekes, R. Teodorescu, M. Liserre, R. Mastromauro, and A. Dell'Aquila, "MPPT algorithm for voltage controlled PV inverters," in *Optimization of Electrical and Electronic Equipment, 2008. OPTIM 2008. 11th International Conference on*, 2008, pp. 427-432.
- [8] M. Qiang, S. Mingwei, L. Liying, and J. M. Guerrero, "A Novel Improved Variable Step-Size Incremental-Resistance MPPT Method for PV Systems," *Industrial Electronics, IEEE Transactions on*, vol. 58, pp. 2427-2434, 2011.
- [9] H. Guan-Chyun, I. H. Hung, T. Cheng-Yuan, and W. Chi-Hao, "Photovoltaic Power-Increment-Aided Incremental-Conductance MPPT With Two-Phased Tracking," *Power Electronics, IEEE Transactions on*, vol. 28, pp. 2895-2911, 2013.

-
- [10] A. Safari and S. Mekhilef, "Simulation and Hardware Implementation of Incremental Conductance MPPT With Direct Control Method Using Cuk Converter," *Industrial Electronics, IEEE Transactions on*, vol. 58, pp. 1154-1161, 2011.
- [11] L. Fangrui, D. Shanxu, L. Fei, L. Bangyin, and K. Yong, "A Variable Step Size INC MPPT Method for PV Systems," *Industrial Electronics, IEEE Transactions on*, vol. 55, pp. 2622-2628, 2008.
- [12] Mellit, H. Rezzouk, A. Messai, and B. Medjahed, "FPGA-based real time implementation of MPPT-controller for photovoltaic systems," *Renewable Energy*, vol. 36, pp. 1652-1661, 2011.
- [13] A. Messai, A. Mellit, A. Massi Pavan, A. Guessoum, and H. Mekki, "FPGA-based implementation of a fuzzy controller (MPPT) for photovoltaic module," *Energy Conversion and Management*, vol. 52, pp. 2695-2704, 2011.
- [14] E.Mohammed, Z. Bashar, and D. J. Atkinson, "Assessment of the Incremental Conductance Maximum Power Point Tracking Algorithm", *sustainable energy, IEEE transactions on*, vol. 4, pp.108-117, 2013.
- [15] Soren Baekhoj Kjaer, "Evaluation of the Hill Climbing and the Incremental Conductance Maximum Power Point Trackers for Photovoltaic Power Systems", *energy conversion, IEEE transactions on*, vol. 27, 2012.
- [16] L.Wu, Z.Zhengming, and L. Jianzheng, "A Single-Stage Three-Phase Grid-Connected Photovoltaic System With Modified MPPT Method and Reactive Power Compensation", *energy conversion, IEEE transactions on*, vol. 22, pp 881-886, 2007.
- [17] D. Lalili, A. Mellit, N. Lourci, B. Medjahed and E.M. Berkouk, "Input output feedback linearization control and variable step size MPPT algorithm of a grid-connected photovoltaic inverter", *Journal Renewable Energy*, pp 3282-3291, 2011
- [18] G.J. Kish J.J. Lee P.W. Lehn, "Modelling and control of photovoltaic panels utilizing the incremental conductance method for maximum power point tracking", *IET Renewable Power Generation*, pp 259-266, 2011.
- [19] J.L.Silva Perales, "Cargador de Baterias a partir de Celdas Solares basado en el convertidor CUK", Tesis, Departamento de Ingeniería Electrónica, Centro Nacional de Investigación y Desarrollo Tecnológico, 2015.
- [20] Líneas de investigación en Electrónica de Potencia, <http://www.cenidet.edu.mx/subaca/web-elec/investigacion/index.html>

Implementación para el algoritmo INC

En este capítulo se detalla primero el comportamiento del algoritmo INC, tanto matemáticamente como por medio de su diagrama de flujo, y se muestran las curvas características del panel a diferentes irradiancias, observándose el comportamiento del algoritmo. Posteriormente se presenta el convertidor *Push-Pull* para el cual se generarán dos señales *PWM*, y el funcionamiento de éste ante un voltaje constante. Se presentan las características básicas y configuraciones del FPGA Spartan 3E. Se detallan las configuraciones tanto de software como de hardware necesarias para realizar la lectura de los voltajes analógicos a la entrada, así como los zócalos para mostrar la salida del FPGA al osciloscopio.

2.1 Detalle del algoritmo

2.1.1 Algoritmo conductancia incremental (INC) con paso fijo

El método INC se basa en la observación del MPP, el cual ocurre bajo la siguiente condición:

$$\frac{dP}{dV} = \frac{d(V.I)}{dV} = 0 \quad (2.1)$$

Al tenerse en cuenta la interdependencia entre la corriente y el voltaje FV, es posible expresar la condición (2.1) como sigue:

$$I + V \cdot \frac{dI}{dV} = 0 \qquad -\frac{I}{V} = \frac{dI}{dV} \qquad (2.2)$$

Lo anterior significa que, en el MPP, el valor absoluto de la conductancia puede ser igual al valor de la conductancia incremental. Esta condición es la base del método de INC y se verifica a través de dos mediciones sucesivas para calcular la conductancia. Como consecuencia, el método requiere la aplicación de la perturbación repetida del valor del voltaje, hasta que ocurra la siguiente condición:

$$\frac{I(i)}{V(i)} = -\frac{I(i) - I(i-1)}{V(i) - V(i-1)} \qquad (2.3)$$

donde los subíndices (i) y $(i-1)$ se refieren a dos muestras consecutivas de voltaje y corriente.

La condición (2.3) se cumple solamente para un sistema ideal; en un caso real los efectos del ruido relacionados con la conversión A/D y el microcontrolador en el cual se implementa el algoritmo impiden que se satisfaga (2.3) de manera exacta. Como una posible consecuencia, el método continuaría entonces comprobando la validación de (2.3) aun en condiciones de irradiancia estacionaria, de modo que las ventajas teóricas del INC sobre otras técnicas se anulan [1], a menos que se ejerza alguna acción correctiva.

El diagrama de flujo del método INC se muestra en la figura 2.1. Periódicamente se miden el voltaje y la corriente. Sean $V(i)$ e $I(i)$ el voltaje y la corriente medidos en un tiempo dado. La variación de la corriente ΔI corresponde a la corriente actual $I(i)$ menos la corriente anterior $I(i-1)$ y ΔV es la diferencia del voltaje actual $V(i)$ con respecto al voltaje anterior $V(i-1)$. Tanto el voltaje como la corriente anteriores $(i-1)$ se refieren a las lecturas en un tiempo anterior. En este trabajo la

medición del voltaje anterior con respecto al actual es de 1 segundo, con fines prácticos. Es decir, para poder capturar la imagen del panel frontal tanto del voltaje actual y como el voltaje anterior.

En principio, la modificación al ciclo de trabajo ΔD es de acuerdo a la relación de la variación de potencia ΔP sobre la variación de voltaje ΔV . Si la relación es igual a cero entonces se está en el MPP. Resulta entonces que no se puede tomar una decisión cuando la tensión se mantiene sin cambios (lo cual implica que $\Delta V = 0$, porque en este caso la relación $\Delta P/\Delta V$ tiende a infinito). Por lo tanto, es preferible establecer ΔD considerando la conductancia y la conductancia incremental en lugar de ΔP y ΔV , como se muestra en la Tabla 2.1.

Tabla 2.1. Método INC

En MPP	$dP/dV = 0$	$\Delta I/\Delta V = - I/V$
A la izquierda del MPP	$dP/dV > 0$	$\Delta I/\Delta V > - I/V$
A la derecha del MPP	$dP/dV < 0$	$\Delta I/\Delta V < - I/V$

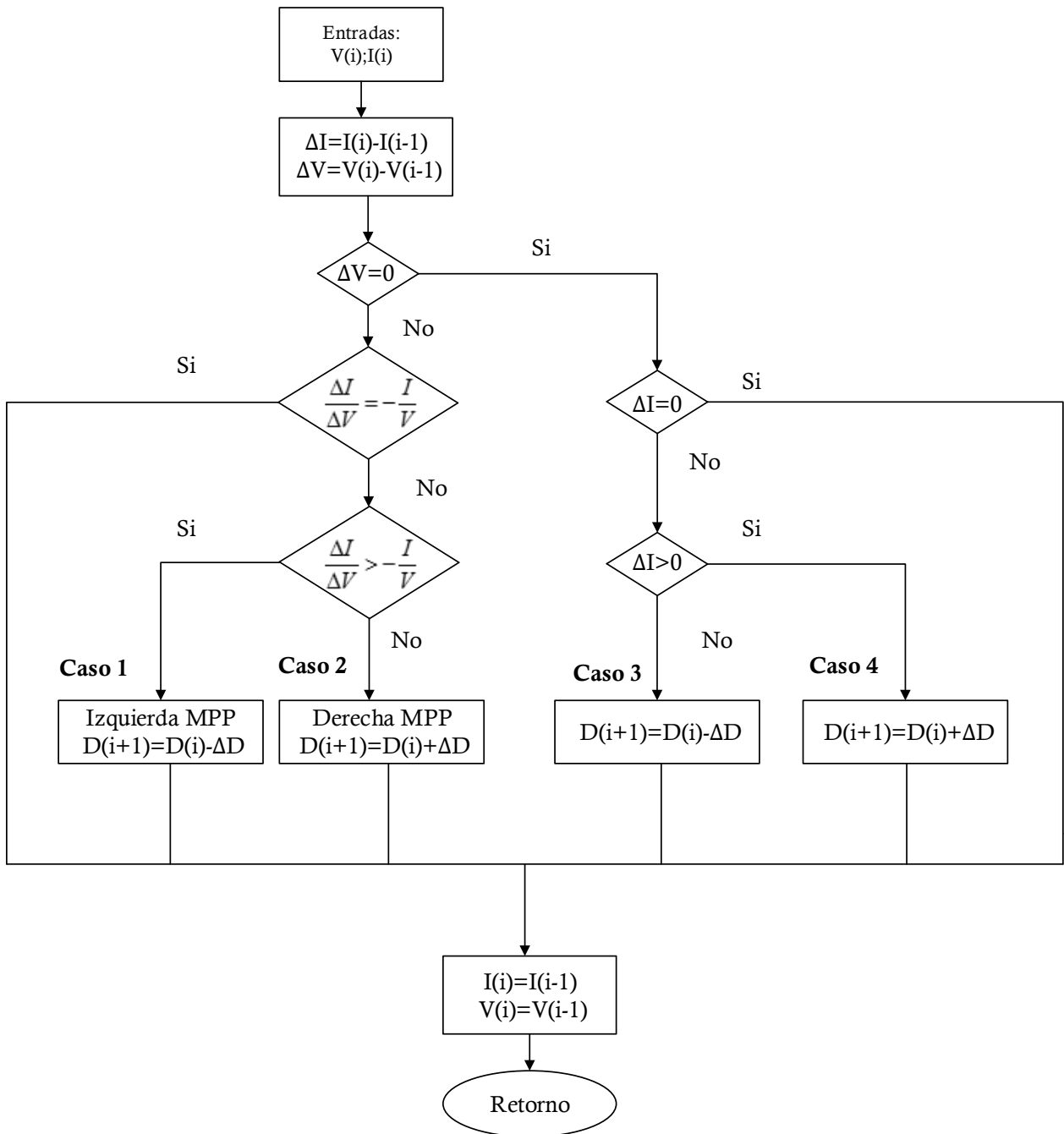


Figura 2.1 Diagrama de flujo del método INC.

Además, el método INC tiene una ventaja potencial: la suma de la conductancia y la conductancia incremental pueden indicar la proximidad del MPP; de hecho, esta suma se mantiene sin cambios si se calcula lejos de MPP, tiende rápidamente a cero en la proximidad de la MPP, y es cero en el MPP [2].

2.1.2 Funcionamiento del algoritmo

A continuación se muestra el comportamiento del algoritmo INC con respecto a la curva de un panel FV a diferentes valores de irradiancias, explicándose los cuatros casos del algoritmo (figura 2.1).

Caso 1: izquierda del MPP

$$\Delta V \neq 0, \quad \frac{\Delta I}{\Delta V} > -\frac{I}{V}, \quad D(i+1) = D(i) - \Delta D \quad (2.4)$$

Con respecto a la figura 2.2, se supone que inicialmente el panel FV opera en el punto A (figura 2.2), y que se produce un cambio muy rápido en la irradiancia y las condiciones climáticas, cambiando al punto B. La corriente actual $I(i)$ disminuye con respecto a la anterior $I(i-1)$, de manera similar el voltaje actual $V(i)$ aumenta con respecto al anterior $V(i-1)$. El punto B se localiza a la izquierda del MPP por lo cual se tendrá que disminuir el ciclo de trabajo para alcanzar este punto, como lo indica la ecuación 2.4.

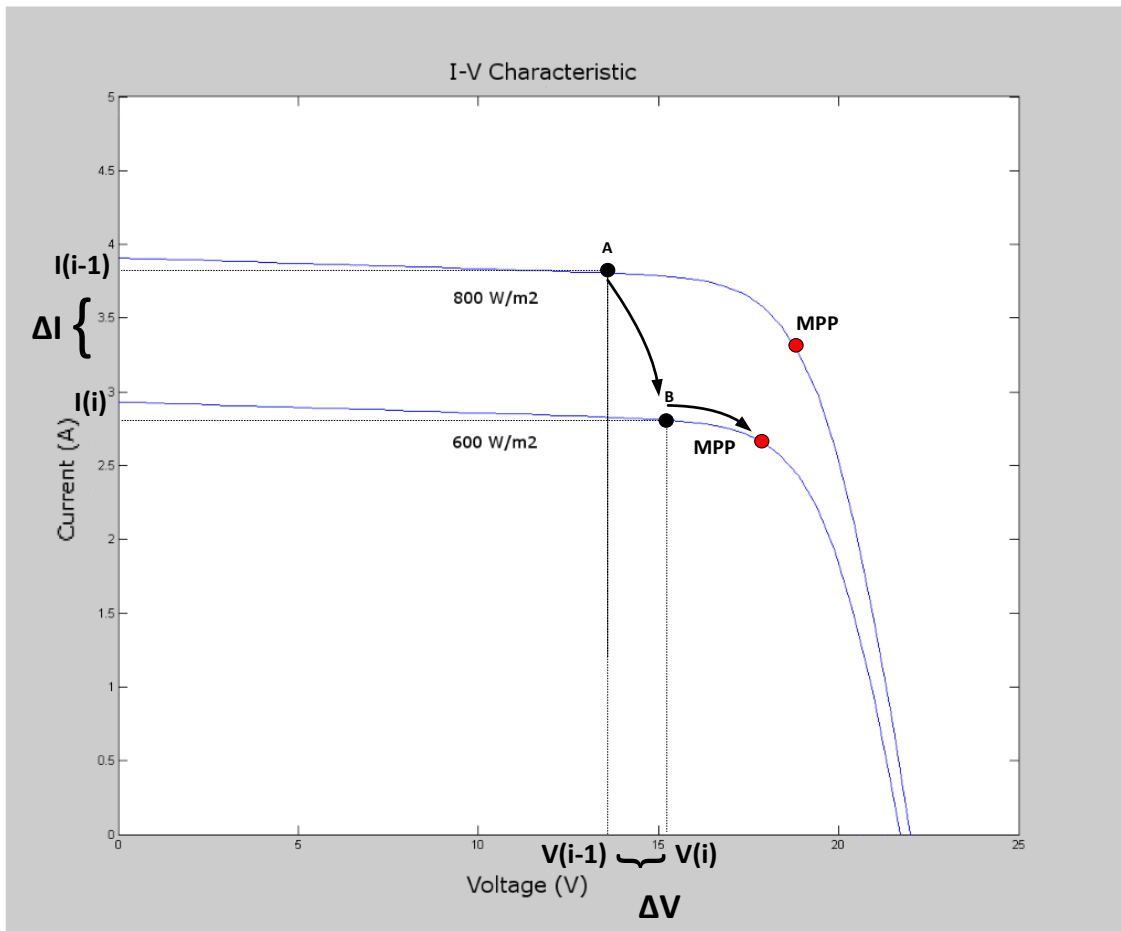


Figura 2.2 Caso 1: izquierda del MPP

Caso 2: derecha del MPP

$$\Delta V \neq 0, \quad \frac{\Delta I}{\Delta V} < -\frac{I}{V}, \quad D(i+1) = D(i) + \Delta D \quad (2.5)$$

En la figura 2.3 se asume que el panel FV opera inicialmente en el punto A. Debido a un aumento muy rápido en la irradiancia y la temperatura, el punto de operación cambia al punto B. La corriente actual $I(i)$ aumenta con respecto a la corriente anterior $I(i-1)$, y el voltaje actual $V(i)$ disminuye con respecto al anterior $V(i-1)$. Como se puede observar, el punto B se encuentra a la

derecha del MPP; como se indica en la ecuación 2.5, el ciclo de trabajo aumentará para alcanzarlo.

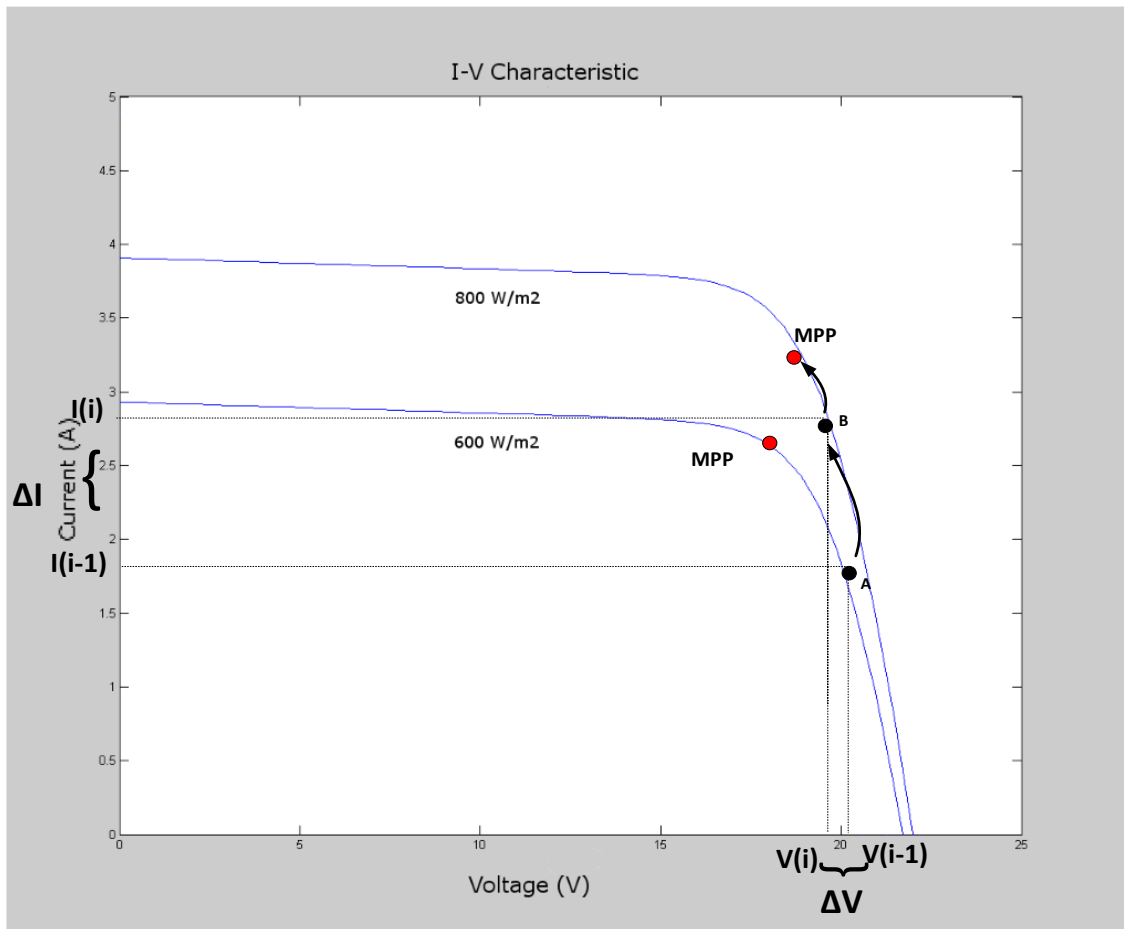


Figura 2.3 Caso 2: derecha del MPP

Caso 3: disminución de irradiancia

$$\Delta V=0, \quad \Delta I<0, \quad D(i+1)= D(i)- \Delta D \quad (2.6)$$

Se asume que el panel FV opera inicialmente en el punto A (figura 2.4). Debido a una disminución en la irradiancia y la temperatura, la operación cambia al punto B. La corriente actual $I(i)$ disminuye con respecto a la corriente anterior $I(i-1)$, no se presentan cambios en el

voltaje actual $V(i)$ con respecto al voltaje anterior $V(i-1)$, por lo que $\Delta V=0$. El ciclo de trabajo disminuirá para alcanzar el MPP, como se indica en la ecuación (2.6). Si se asume que el panel FV opera en el punto C, de igual manera la $\Delta I < 0$, y la $\Delta V=0$, el ciclo disminuirá desplazándose a la derecha del MPP. Al realizar la siguiente lectura el ΔV será diferente de cero y el algoritmo identificará que se encuentra al lado derecho del MPP, realizando la corrección correspondiente (Caso 2).

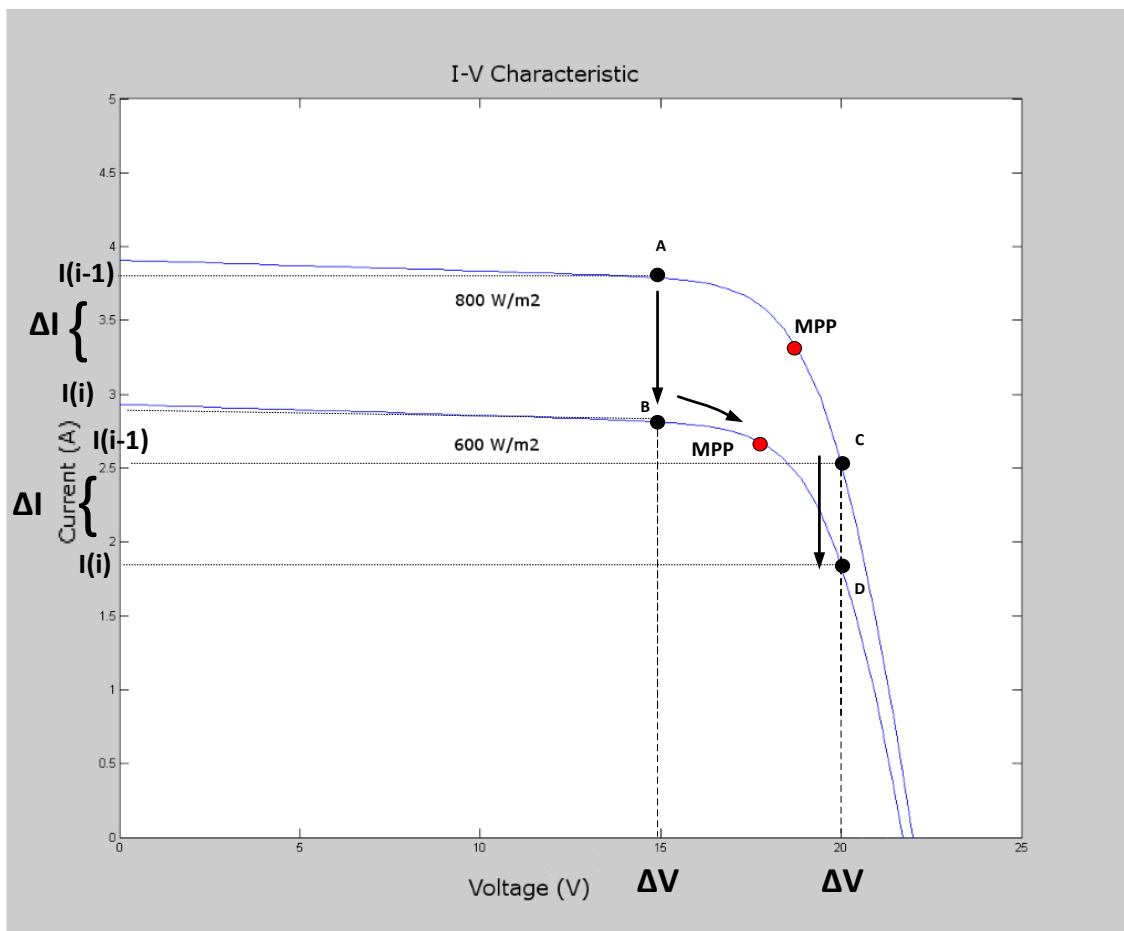


Figura 2.4 Caso 3: disminución de irradiancia

Caso 4: aumento de irradiancia

$$\Delta V=0 \qquad \Delta I>0 \qquad D(i+1)=D(i)+\Delta D \qquad (2.7)$$

Se asume que el panel FV opera inicialmente en el punto A (figura 2.5) y que, debido a un aumento en la irradiancia y la temperatura, cambia al punto B. La corriente actual $I(i)$ aumenta con respecto a la corriente anterior $I(i-1)$, no se presenta cambio del voltaje actual $V(i)$ con respecto al voltaje anterior $V(i-1)$, siendo $\Delta V=0$. Como consecuencia el ciclo de trabajo aumentará acercándose al MPP como se indica en la ecuación 2.7. Si se asume que el panel FV opera inicialmente en el punto C, por un aumento de irradiancia el $\Delta I>0$, y no se presentan cambios en el voltaje ($\Delta V=0$), el ciclo de trabajo aumentara desplazándose a la izquierda del MPP. Al realizar la siguiente lectura, el algoritmo identificará que se encuentra al lado izquierdo del MPP realizando la corrección necesaria (Caso 1).

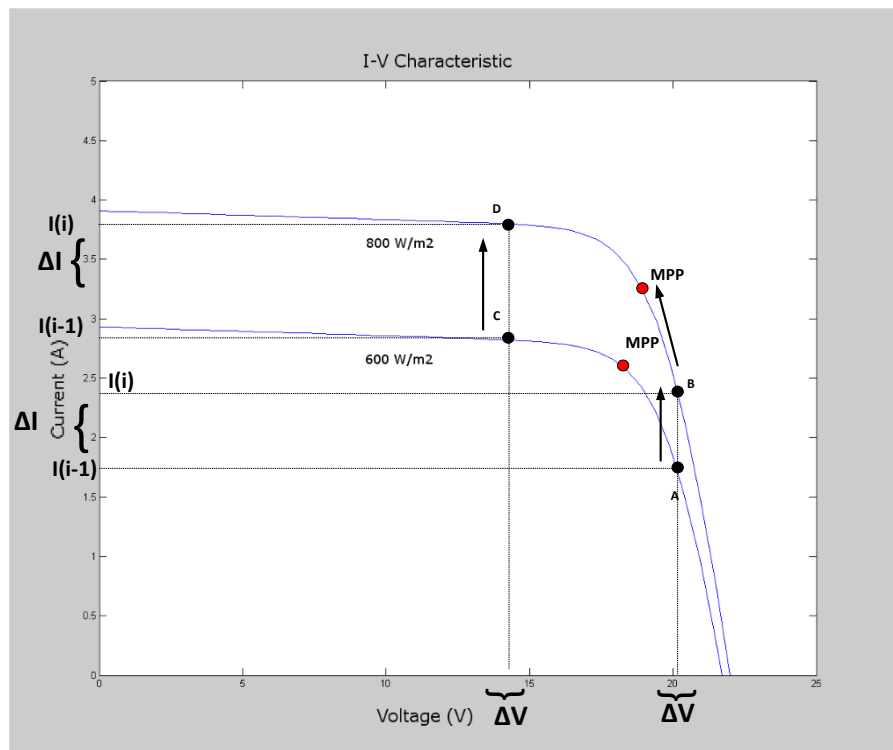


Figura 2.5 Caso 4: aumento de irradiancia

2.1.3 Sistema para el control de MPP.

En la figura 2.6 se muestra el diagrama a bloques del sistema para el cual se diseñó el algoritmo de seguimiento del MPP, implementado en el FPGA Spartan 3E.

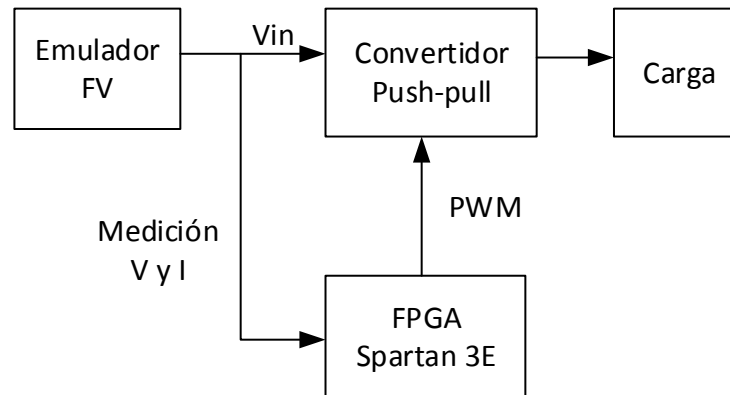


Figura 2.6 Sistema general para la implementación del FPGA

En este sistema se emplea el emulador fotovoltaico (Magna-power electronics) de 10 kW. El convertidor para el cual se generarán las señales es un Push-Pull (figura 2.7), debido a que éste es utilizado ampliamente en aplicaciones de bajo voltaje y elevada corriente de entrada, como las fotovoltaicas, además proporciona aislamiento galvánico [3]. Se desarrollaron cuatro prototipos en la tesis [3]. El sistema se considera aislado de la red eléctrica y la carga se mantiene constante.

2.1.3.1 El funcionamiento del Push-Pull.

El máximo ciclo de trabajo de este convertidor es ligeramente menor al 50% ya que, si se superará este valor, se presentaría el disparo simultáneo de ambos interruptores, provocando un corto circuito. Para evitar esto es necesario incluir un tiempo muerto entre el tiempo de apagado

de un interruptor y el tiempo de encendido del otro. Además, los interruptores funcionan desfasados 180° . A continuación se describe su funcionamiento.

El ciclo de operación de este convertidor se divide en tres etapas:

- Etapa 1: Cuando Q1 está cerrado y Q2 está abierto, el voltaje de entrada (V_{in}) se aplica a la mitad del devanado primario del transformador.
- Etapa 2: El interruptor Q2 se encuentra cerrado y Q1 abierto, V_{in} se aplica a la otra mitad del devanado primario y la dirección de la corriente desmagnetiza el transformador.
- Etapa 3: Ambos interruptores se abren, los dos diodos de libre circulación conducen simultáneamente la corriente del inductor de salida.

Las formas de onda características de este convertidor se muestran en la figura 2.8 [3].

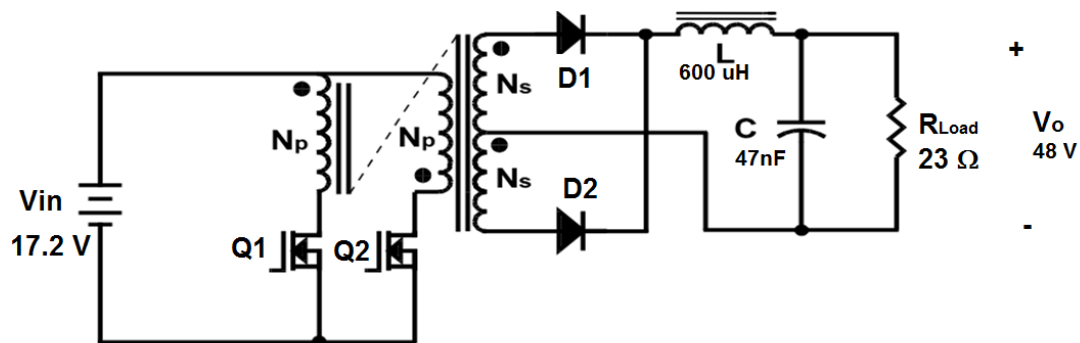


Figura.2.7 Convertidor Push-Pull

El push-pull diseñado es de 100W, debido a que los convertidores utilizados para un panel FV no conectados a la red típicamente tienen estas potencias. El voltaje de entrada es con referencia al voltaje de MPP del panel FV Conergy C125W y el voltaje de salida de 48V ya que es voltaje estándar para conectar bancos de 4 baterías. Sus características son las siguientes:

- $P=100\text{ W}$
- $I_o=2.083\text{ A}$
- $V_o=48\text{ V}$
- $D=0.32$
- $f_s=100\text{ kHz}$
- $R_L=23\ \Omega$
- $V_{in}=17.2\text{ V}$
- $L=600\ \mu\text{H}$
- $C=47\text{ nF}$

2.2 Detalle del hardware

2.2.1 FPGA Spartan 3E.

En esta sección se abordan algunas características básicas del FPGA, describiéndose también la configuración necesaria para descargar directamente, por medio del puerto USB (Universal Serial Bus), el programa realizado en Labview. Se describe también la configuración tanto de hardware como software para llevar a cabo la lectura del voltaje analógico en los dos canales. La tarjeta Spartan-3E (ver figura 2.9), contiene varios componentes; los más importantes para este trabajo son [4]:

Modelo FPGA Xilinx XC3S500E.

- 500,000 arreglos de compuertas
- 4 Mbit de memoria PROM (Programmable Read-Only Memory)
- 64 MB (512 Mbit) de memoria DDR SDRAM (Double Data Rate Synchronous Dynamic Random Access Memory)
- 16 MB (128 Mbit) de memoria paralela tipo NOR Flash
- Reloj de 50 MHz
- Seis conectores de expansión de seis terminales
- Cuatro canales para conversión digital analógica matrícula LTC2664

- Dos canales para conversión analógica a digital matrícula LTC1407 con preamplificador de ganancia programable matrícula LTC6912-1
- Ocho LEDs para señalización

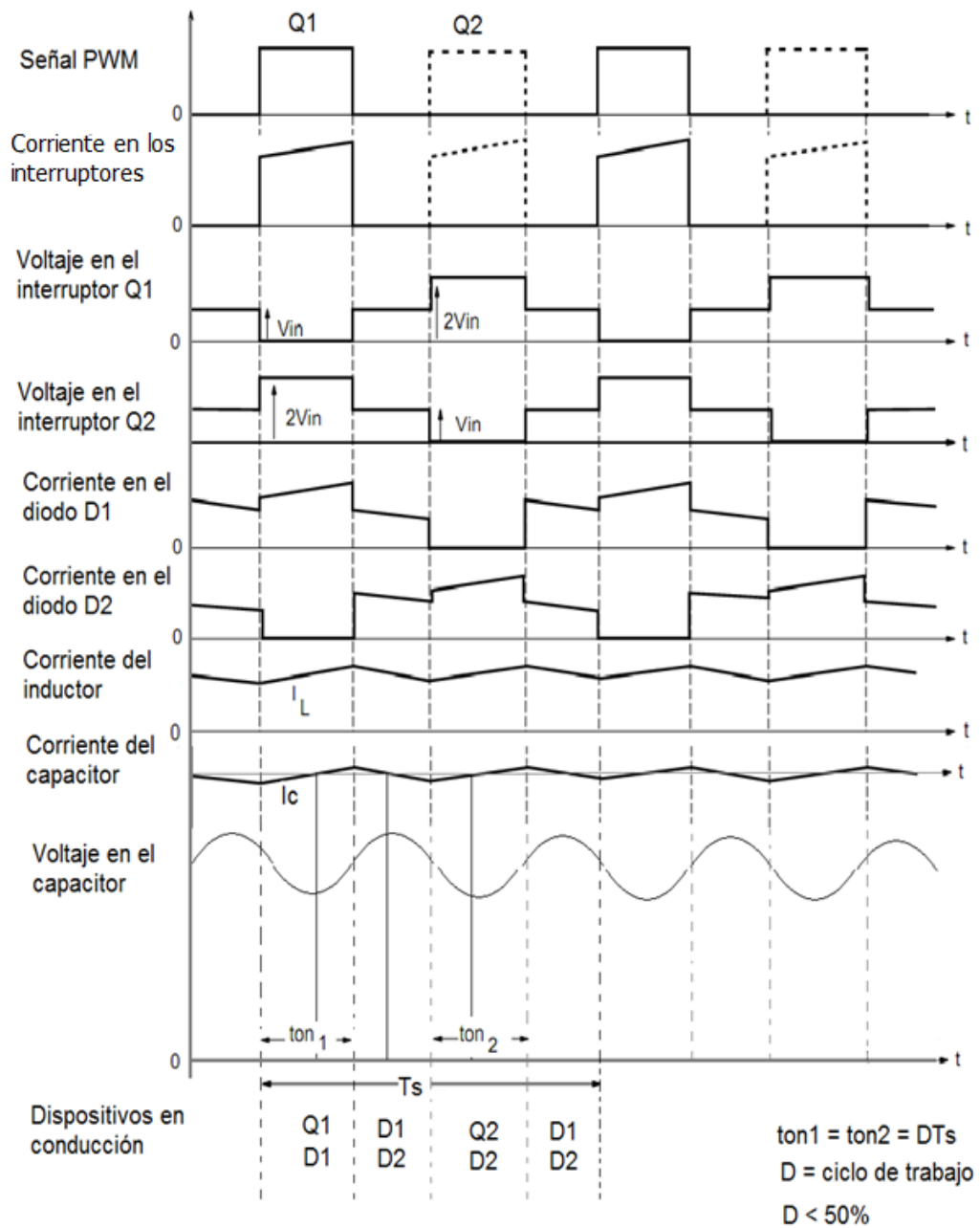


Figura 2.8 Formas de Onda del convertidor Push-Pull



Figura 2.9 FPGA Spartan 3E

2.2.1.1 Opciones de configuración del FPGA

La tarjeta Spartan-3E acepta una variedad de configuraciones. Una de ellas, la que se usará en el presente trabajo, es descargar el diseño directamente al FPGA vía JTAG (Joint Test Action Group), usando la interfaz del puerto USB de la tarjeta. Esta configuración se selecciona a través del Jumper 30, el cual determina cuál modo de configuración usará el FPGA cuando se alimente por primera vez o cuando se presione el botón PROG. El Jumper 30 deberá estar como se muestra en la figura 2.10, para que la configuración sea a través del puerto USB [4-5].

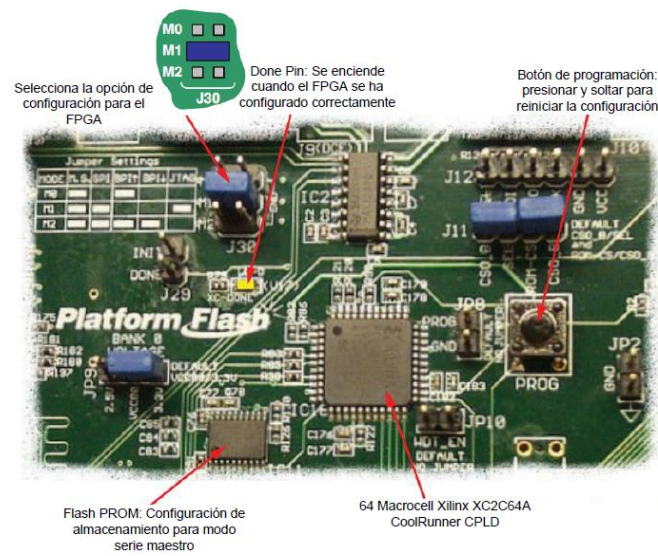


Figura 2.10 Opciones de configuración detallada

2.2.1.2 Expansión de conectores

La tarjeta Spartan-3E provee conectores de expansión para facilitar la conexión con otros dispositivos. Las diferentes conexiones de entrada/salida se observan en la Figura 2.11. El conector de interés en el presente trabajo es el accesorio de salida J1 (6 pines) [2-4].

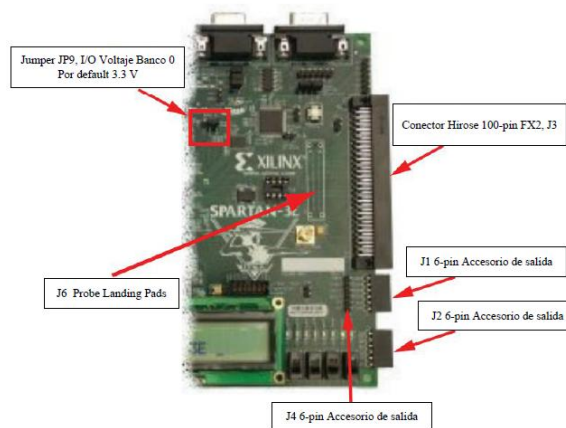


Figura 2.11 Conectores de expansión

2.2.1.3 6 pines de accesorio de salida (Accessory Headers)

El header J1 se muestra en la figura 2.12; son 6 pines a lo largo del borde derecho de la tarjeta. Usa 6 pines hembras a 90°. Se conectan cuatro pines de FPGA al header J1, FX2_IO <4:1>. Estas cuatro señales también se comparten con el conector Hirose FX2. La tarjeta suministra 3.3 V para una tarjeta externa montada en el zócalo J1 sobre el pin inferior [2-4].

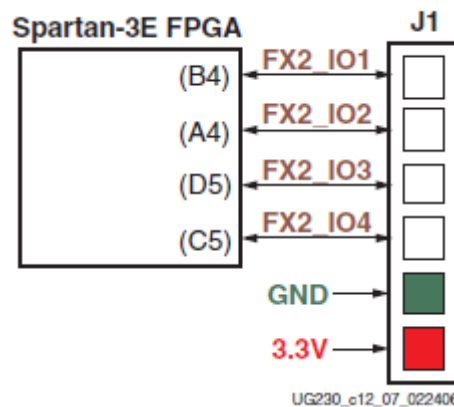


Figura 2.12 conexiones del FPGA al J1 Accessory Header.

2.2.1.4 Módulo ADC (Convertidor analógico-digital) en Spartan 3E FPGA.

La tarjeta Spartan-3E incluye una etapa de captura analógica de dos canales (figura 2.13), el cual incluye un circuito pre-amplificador programable (LTC6912-1) para escalar la señal analógica de entrada. La salida del pre-amplificador se conecta con el ADC (LTC1407-1). Ambos circuitos los programan en serie por medio de la interfaz SPI (Serial Peripheral Interface) del FPGA.

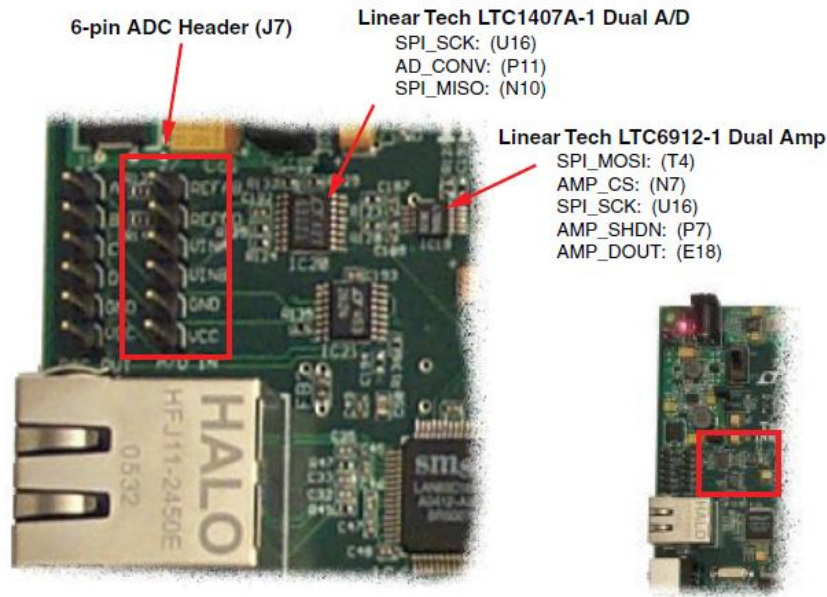


Figura 2.13 Canales de captura del circuito analógico.

La tarjeta proporciona una interfaz periférica serial (SPI). El bus SPI es full-duplex con comunicación serial sincrónica, y establece un esquema maestro-esclavos manejado por 3 señales:

1. MOSI (Master Output, Slave Input),
2. MISO (Master Input, Slave Output)
3. SCK (Reloj del Sistema).

En este caso, el maestro es el FPGA y los esclavos son el DAC LTC2624, el ADC LTC1407A-1 y el pre-amplificador LTC6912-1. Puesto que los esclavos comparten el bus SPI, además de estas tres señales cuentan con señales individuales para su habilitación, de manera que no existan conflictos para un dispositivo mientras otro se está utilizando.

La interfaz SPI basa su operación en registros de desplazamiento. El maestro determina la velocidad y marca el inicio de la transferencia. Los datos recibidos por el esclavo se devuelven para su verificación. Esto se ilustra en la figura 2.14, donde la señal *ss* es para seleccionar un esclavo en particular, dado que pueden ser varios los que comparten la interfaz [4,6].

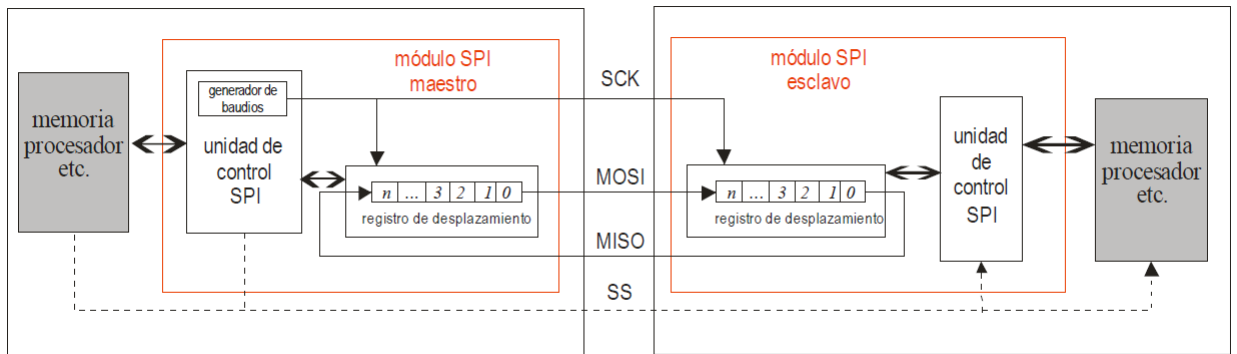


Figura 2.14 Comunicación via SPI

Convertidor Análogo Digital (ADC).

Este ADC está incorporado en la placa del FPGA y es un integrado de Linear Technology cuya matrícula es LTC1407A. Es un ADC de dos canales, cuenta con una resolución de 14 bits, con voltaje de referencia de 1.65 V por canal y una frecuencia de muestreo de hasta 1.5MHZ. Cada canal del ADC acepta entradas entre 0.4 V y 2.9 V, motivo por el cual es necesario que en su entrada se adecúe el voltaje de la señal mediante algún circuito sujetador.

Las muestras digitales de las señales de entrada, de acuerdo a la conversión que realiza el ADC, corresponden a valores entre -8192 y 8192, contenidos en los 14 bits en representación de complemento a dos que se entregan en formato serie en una misma línea del ADC para ambos canales.

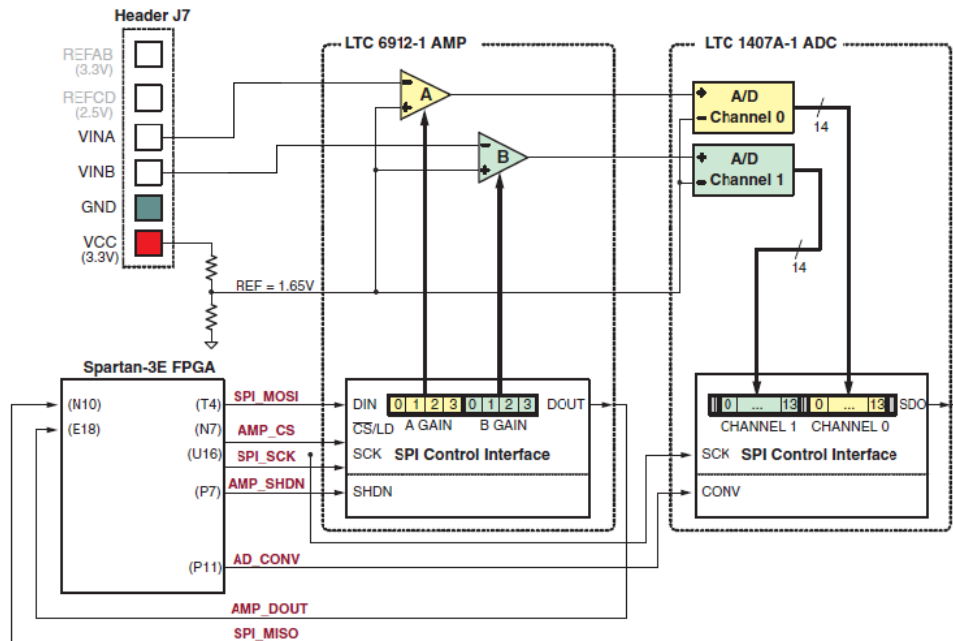


Figura 2.15 Circuito de captura de señales analógicas.

Como se podrá observar en la figura 2.15, el ADC va acompañado de un preamplificador programable de ganancia variable de matrícula LTC6912. Los dispositivos se comunican mediante el bus SPI de la tarjeta (además de otros dispositivos SPI como un DAC y memoria FLASH), por lo cual se debe tener especial cuidado en deshabilitar los otros dispositivos mientras se utilizan los buses de comunicación [4].

Preamplificador programable

El pre-amplificador LTC6912 consta de 2 amplificadores de ganancia programable de poco ruido, fáciles de utilizar y que ocupan muy poco espacio. Las señales analógicas se conectan a los pines *VinA* o *VinB* del conector J7, con un voltaje de referencia $V_{ref}=1.65$ V. El 1.65 V de referencia se genera mediante un divisor de tensión a partir de los 3.3 V de la tensión de alimentación. La ganancia de cada amplificador es programable de -1 a -100, como se muestra en la tabla 2.2. Las ganancias para los dos canales se programan de manera independiente utilizando

la interfaz SPI. Todas las ganancias están invertidas. En este caso se utiliza con una ganancia de -1 dado que es la que ofrece el rango de entrada más amplio [2-4].

Tabla 2.2 Ganancias del amplificador.

Gain	A3	A2	A1	A0	Input Voltaje Range	
	B3	B2	B1	B0	Min	Max
0	0	0	0	0		
-1	0	0	0	1	0.4	2.9
0.2	0	0	1	0	1.025	2.275
-5	0	0	1	1	1.4	1.9
-10	0	1	0	0	1.525	1.775
-20	0	1	0	1	1.5875	1.7125
-50	0	1	1	0	1.625	1.625
-100	0	1	1	1	1.6375	1.6625

SPI Control Interface

En la figura 2.16 se muestra la interfaz de comunicaciones SPI con el amplificador. La ganancia de cada amplificador se envía como una palabra de 8 bits de comando, que consta de dos campos de 4 bits. El bit más significativo, B3, se envía primero [2-4]. La salida AMP_DOUT del amplificador contiene la información de la ganancia. Estos valores pueden ser ignorados en la mayoría de las aplicaciones.

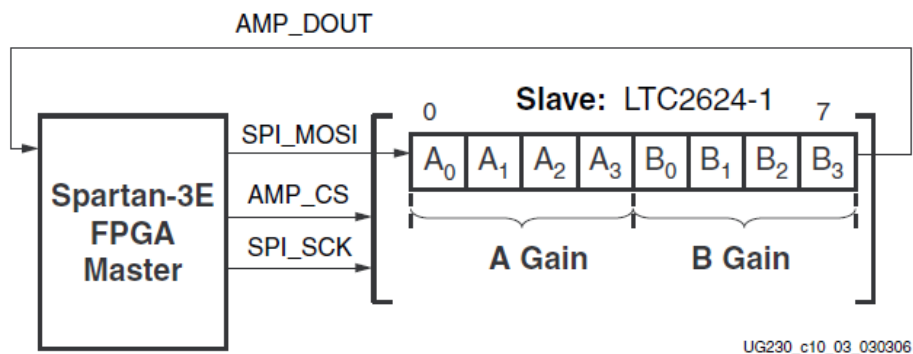


Figura 2.16 Interface serial SPI para el amplificador

La transacción del bus SPI se inicia cuando la FPGA coloca en bajo la señal AMP_CS (figura 2.17). El amplificador captura los bits en serie de SPI_MOSI en el flanco de subida de la señal de reloj SPI_SCK. El amplificador presenta los bits en serie AMP_DOUT en el flanco de bajada de la señal SPI_SCK.

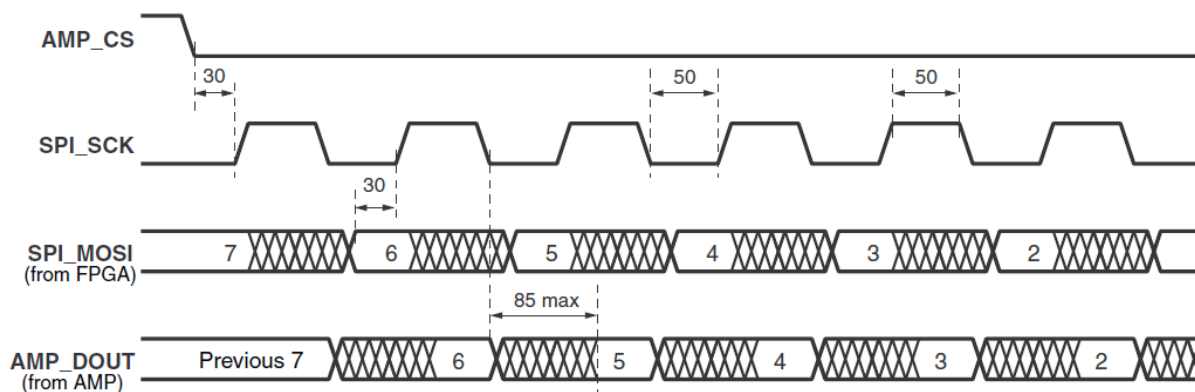


Figura 2.17 Diagrama de tiempos para la comunicación con el amplificador

La configuración se envía por la línea SPI_MOSI (Master Out Slave In) empezando por el bit B3 y terminando por el bit A0. El registro de configuración actualiza sus valores a medida que los va recibiendo en los flancos de subida del reloj SCK mientras la señal AMP_CS (chip select del amplificador) se encuentre en 0 lógico. La configuración se termina cuando se sube la señal AMP_CS y a partir de ese momento ya se puede utilizar el ADC. La configuración utilizada para el *core* es 00010001, es decir ganancia -1 para ambos canales, de manera de obviar este componente [4].

Salidas digitales de entradas analógicas.

El resultado del circuito de captura (para uno de los canales) corresponde a un valor en complemento a 2 de 14 bits, que en valor decimal se explica de acuerdo a la siguiente fórmula:

$$D[13:0] = (\text{Ganancia}) \left[\frac{(V_{in} - 1.65V)}{1.25V} \right] (8192) \quad (2.4)$$

En (2.4), el voltaje de referencia para el amplificador y el ADC es de 1.65 V, generado por un divisor de voltaje; por lo tanto, se resta del voltaje de entrada. El intervalo de entrada del ADC es ± 1.25 V, centrado alrededor del voltaje de referencia; por consiguiente, aparece en el denominador. Finalmente, el ADC presenta una salida digital de 14 bits en complemento a dos (ver figura 2.18), la cual representa valores entre -2^{13} y $2^{13}-1$ es por esto que la cantidad queda escalada por 8192 [2-6].

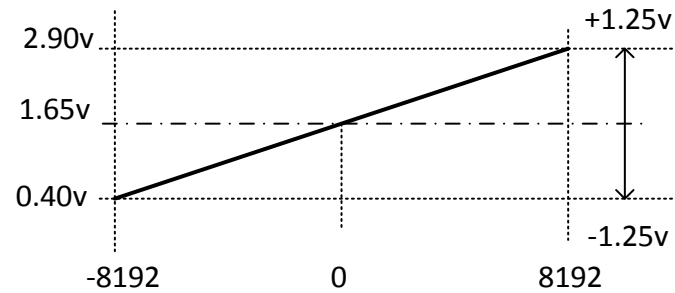


Figura 2.18 Representación de la salida del ADC.

Interface de control SPI del ADC

El proceso de conversión inicia cuando se envía un pulso por la línea AD_CONV. En este instante ambos canales analógicos se muestrean simultáneamente; el resultado de la conversión se obtiene hasta que se manda otro pulso igual al que la inició. El proceso de transacción en el bus SPI se observa en la Figura 2.19. La señal AD_CONV no es la tradicional que habilita el dispositivo esclavo en una comunicación SPI. Por otro lado, se deben asegurar los suficientes ciclos de reloj para que el ADC deje la línea SPI_MISO en alta impedancia.

Como ya se mencionó, el resultado de la conversión es una representación digital en valores de 14 bits en complemento a dos, el bit 13 es el de signo. Finalmente, como las señales del bus SPI se comparten con otros dispositivos en la tarjeta, es de suma importancia que estos dispositivos estén deshabilitados cuando se requiera comunicar el FPGA con el amplificador o con el ADC (ver Tabla 2.3) [4].

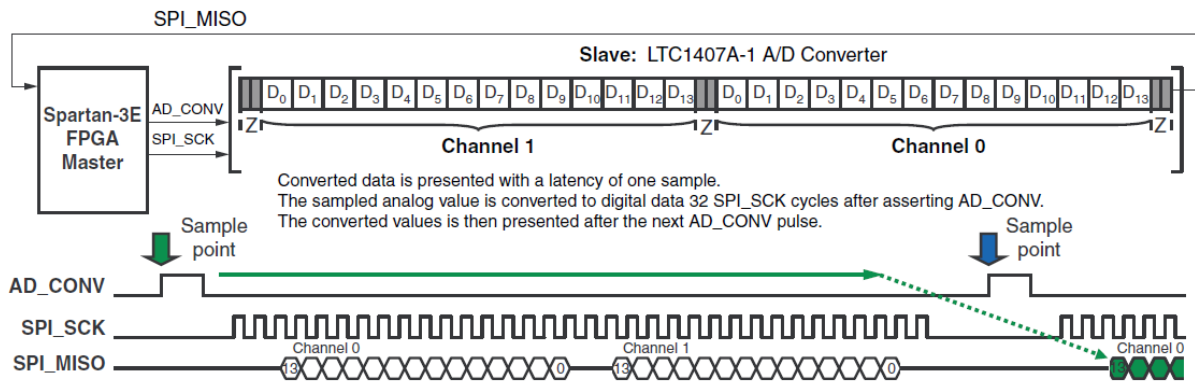


Figura 2.19 Interface de conversión analógico a digital

Tabla 2.3 Líneas para deshabilitar otros dispositivos en el bus SPI.

Señal	Dispositivo a deshabilitar	valor
SPI_SS_B	SPI Serial Flash	1
AMP_CS	Preamplificador programable	1
DAC_CS	DAC	1
SF_CE0	StrataFlash Parallel Flash PROM	1
FPGA_INIT_B	Platform Flash PROM	0

2.2.2 Programación gráfica Labview

Labview es el acrónimo de *Laboratory Virtual Instrument Engineering Workbench*. Es un lenguaje y a la vez un entorno de programación gráfica en el que se pueden crear aplicaciones de una forma rápida y sencilla. Originalmente este programa estaba orientado a aplicaciones de control de instrumentos electrónicos usadas en el desarrollo de sistemas de instrumentación, lo que se conoce como instrumentación virtual. Por este motivo, los programas creados en LabVIEW se guardan en archivos llamados VI y con la misma extensión, que significa instrumento virtual (Virtual Instruments). También relacionado con este concepto se da nombre a sus dos ventanas principales: un instrumento real tendrá un panel frontal donde estarán sus

botones, pantallas, etc., y una circuitería interna. En Labview estas partes reciben el nombre de Panel frontal y Diagrama de bloques respectivamente:

Panel frontal, es la parte que verá el usuario, suele tener el fondo gris.

Diagrama de bloques, es donde se realizará la programación y suele tener fondo blanco.

En muchas ocasiones, un programa será de un tamaño tal que habrá que separarlo en varios ficheros o habrá alguna sección de código que convenga reutilizar varias veces. Un VI puede contener a otro de forma que el segundo sería un subVi del primero, el concepto es equivalente a las funciones de una subrutina en un lenguaje tradicional [7].

2.2.2.1 Estructuras de Labview.

Las instrucciones de control permiten a un programa ejecutar un código de forma condicional o repetirlo cierto número de veces. En Labview estas instrucciones son estructuras que encierran en su interior el código que afectan. Se encuentran en el menú *Programming > Structures*. La estructura *flat sequence* (figura 2.20) sirve para ordenar el orden de la ejecución del código que está en su interior. Esta estructura tiene fotogramas o *frames* ordenados; en el interior de cada *frame* se situará una sección del código. La ejecución comenzará por el primer *frame*, cuando acabe se pasará a ejecutar el siguiente, y así sucesivamente [7].

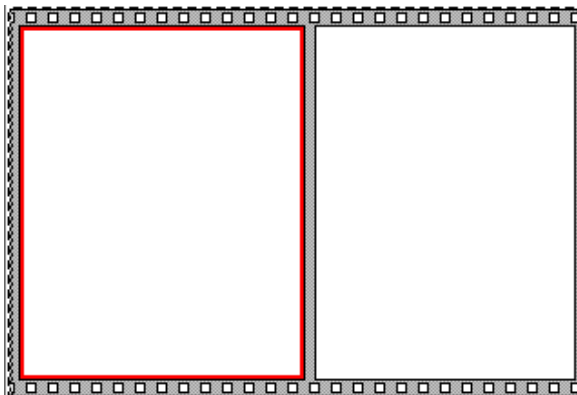


Figura 2.20 Estructura flat sequence.

2.3 Bibliografía

- [1] N.Femia, G.Petrone, G. Spagnuolo, M.Vitelli, “Power Electronics and Control Techniques for Maximum Energy Harvesting in Photovoltaic Systems”, editorial CRC Press, pp 71-144, 2012.
- [2] Menniti, D., Burgio, A., Sorrentino, N., Pinnarelli, A., & Brusco, G., “An incremental conductance method with variable step size for MPPT: Design and implementation”, In Electrical Power Quality and Utilisation, 2009. EPQU 2009. 10th International Conference on (pp. 1-5).
- [3] S.E. De León Aldaco, “Comparación de metodologías de evaluación de confiabilidad”, tesis, Departamento de Ingeniería Electronica, Centro Nacional de Investigación y Desarrollo tecnológico, 2011.
- [4] Xilinx, “Spartan-3E FPGA Starter Kit Board User Guide.” Xilinx, pp. 76-81, 2011.
- [5] Josefa Gordillo Estrada, “Compensación por Medio de la Modulación de la Onda de Salida de un Inversor Multinivel en Cascada ante Desbalances en las Fuentes de Alimentación“, tesis, Departamento de Ingeniería Electronica, Centro Nacional de Investigación y Desarrollo tecnológico, 2009.
- [6] Notas de clase Profesora Lucelly Reyes, “Módulo ADC (Convertidores analógicos-digitales)”, en línea. Disponible en :http://fisica.udea.edu.co/~labgicm/Curso%20de%20Instrumentacion/2013_Clase_FPGA_ADC.pdf
- [7] J.R. Lajara Viazcaínoy J.Pelegrí Sebastián,”Labview entorno gráfico de programación, editorial Alfaomega Marcombo, primera edición, pp 3,4,20, 2007.

Programación en el FPGA

En este capítulo, primero se aborda las limitaciones del módulo de FPGA en Labview, especificando las operaciones y funciones que no se pueden realizar dentro de ésta, lo que llevó a utilizar un método llamado Host, para solucionar este problema. Se explica cómo funciona el método Host y otras aplicaciones para lo cual también se utiliza. Posteriormente se aborda la programación en Labview para la adquisición de voltajes, almacenamiento y su acondicionamiento para llevar a cabo el algoritmo de INC.

3.1 Limitaciones en la paleta del VI del FPGA.

Algunos VIs y funciones de LabVIEW no están disponibles o tienen restricciones para ejecutarse dentro del FPGA; estas funciones son las siguientes:

- Funciones de punto flotante
- Arreglos de tamaño variable y multidimensionales
- Grupos o cadenas de error
- Cajas de diálogo
- Archivos Entrada/Salida
- Impresión

- Punto fijo
- Los datos con formato sencillo y doble no se pueden manejar [1].

3.1.1 Modo PC interactiva (Host)

Se interactuará mediante programación con un VI residente en el FPGA, utilizando un VI separado que se ejecuta en el equipo Host.

Cuando se utiliza la comunicación por medio de la interfaz de programación, el VI del FPGA se ejecuta en la tarjeta del FPGA y el *Host VI* se ejecuta en el equipo Host, como se muestra en la figura 3.1. En la figura 3.2 se muestra la interacción con el FPGA a través del panel frontal basado en la PC Host (anfitrión), lo que permite desarrollar otro proceso en el VI Host.

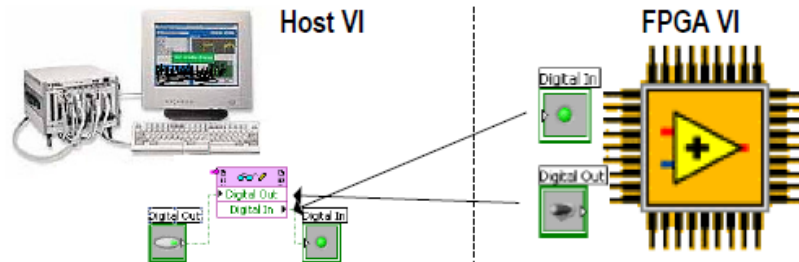


Figura 3.1 Interacción FPGA y Host.

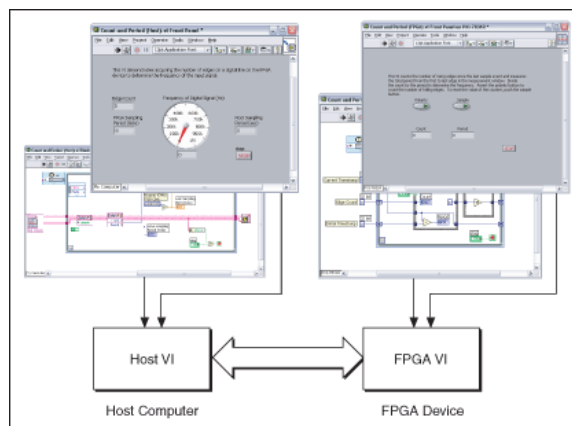


Figura 3.2 Host.

Se usa el *Host VI* para enviar información entre la PC y la tarjeta de FPGA por las siguientes razones:

- Se pueden procesar mas datos de los que caben en el FPGA
- Permite llevar a cabo operaciones que no están disponibles en la tarjeta del FPGA, como de doble precisión o precisión extendida de punto flotante
- Permite crear una aplicación de varios niveles, con la tarjeta FPGA como un componente de un sistema mayor
- Permite Registrar datos
- Es posible controlar el momento y la secuencia de transferencia de datos
- Se puede crear un banco de pruebas para un VI de FPGA [2]

3.2 Programación del FPGA

La figura 3.3 muestra el diagrama de bloques general para llevar a cabo la adquisición de voltaje en los dos canales del FPGA (Canal A y Canal B) y su almacenamiento. Cabe aclarar que las señales de entrada en ambos canales son voltajes analógicos suministrados por una fuente de voltaje, suponiendo las señales ya convenientemente acondicionadas para implementar el algoritmo de INC, es decir realizada la transducción de corriente a voltaje por medio de un sensor hall (canal B).

Las operaciones que se llevan a cabo son las siguientes: el bloque del preamplificador se configura a una ganancia de -1; se lee el voltaje; se guarda la lectura en una localidad de memoria [3] durante un segundo (puede cambiarse según requiera la rapidez del algoritmo).

Posteriormente se lee nuevamente el voltaje analógico, tal como lo requiere el proceso del algoritmo INC; esta segunda lectura corresponde al voltaje actual.

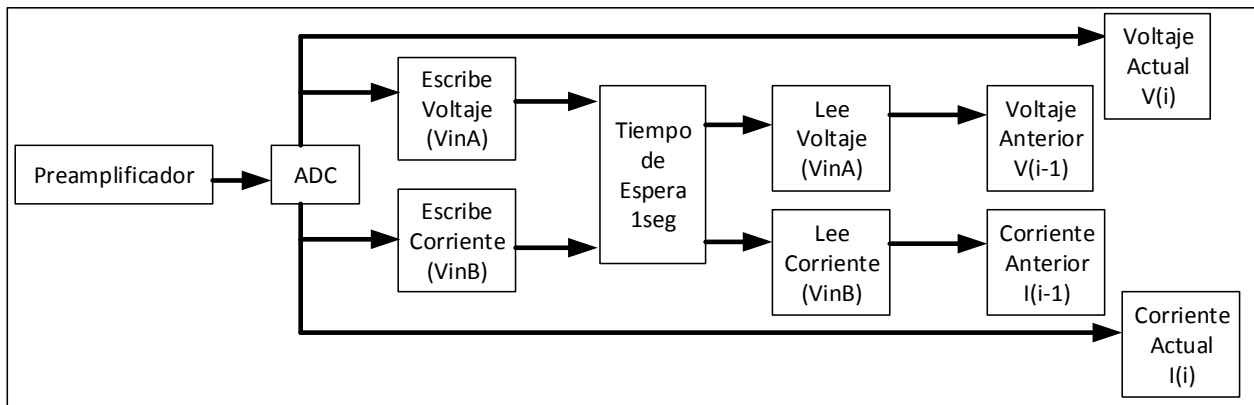


Figura 3.3 Diagrama a bloques para adquisición y almacenamiento del voltaje analógico

3.2.1 Configuración del Preamplificador FPGA.

Las figuras 3.4-3.7 corresponden al bloque del preamplificador de la figura 3.3. Para configurar la ganancia de -1 se utiliza una estructura de Labview llamada *flat sequence*; en ésta, cada segmento del programa se ejecuta secuencialmente de izquierda a derecha. Para esta configuración primero se deshabilitan las siguientes señales del FPGA, de manera que no exista conflicto por el uso del bus SPI (figura 3.4):

```

SPI_SS_B = 1
DAC_CS = 1
SF_CE0 = 1
FPGA_INIT_B = 0
DAC_CLR = 1
STRATAFLASH_OE = 1
STRATAFLASH_WE = 1
  
```

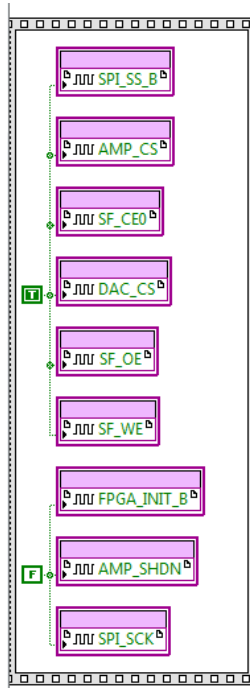


Figura.3.4 Habilitación de señales

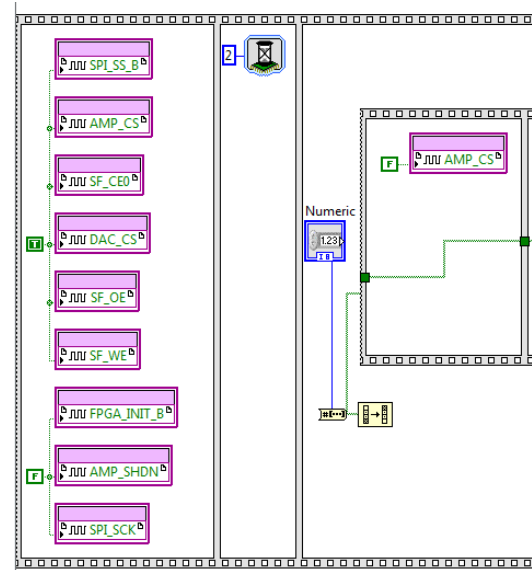


Figura 3.5. Programa preamplificador.



Figura 3.6. Contador hasta 8 para terminar de configurar el preamplificador

La figura 3.6 se repite 8 veces para configurar las ganancias de los canales A y B. La figura 3.7 muestra la parte del programa que permite ver la ganancia programada en los LEDs del FPGA.

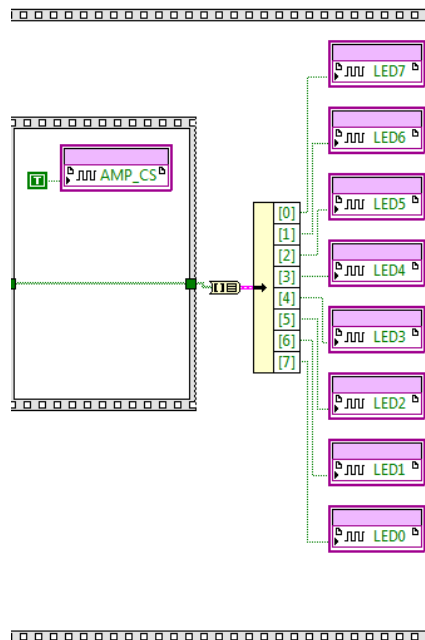


Figura 3.7 Ganancia en LEDs

Se define el subprograma (llamado subVI en Labview) del preamplificador que se muestra en la figura 3.8; esto permite tener el programa de manera compacta y facilita agregar los demás bloques necesarios para el algoritmo INC.



Figura.3.8 SubVI preamplificador

3.2.2 Programa para el ADC

Las figuras 3.9-3.11 corresponden al bloque del ADC de la figura 3.3. Constituyen un solo programa y se ejecutan de izquierda a derecha. Este programa sensa el voltaje de entrada para ambos canales y realiza las operaciones para representarlo digitalmente en el panel frontal.

La figura 3.9 muestra el programa para la adquisición de voltaje analógico a la entrada del FPGA. Se habilitan las señales mostradas en 1 lógico (true) y se comienza a leer la señal analógica. En la figura 3.10 se muestran los 32 pulsos de reloj que se requieren para tomar la lectura de los dos canales analógicos.

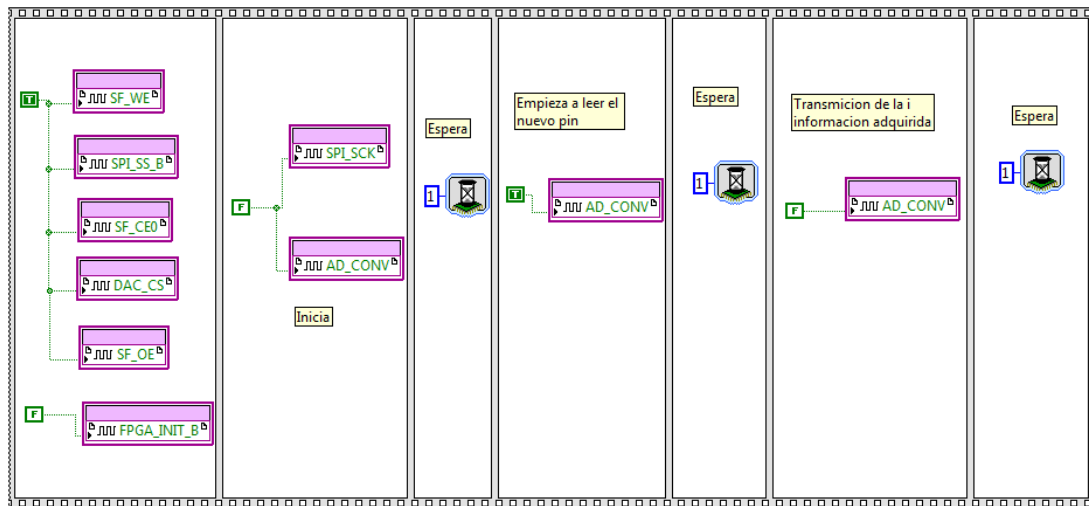


Figura 3.9 Adquisición voltaje analógica (parte I)

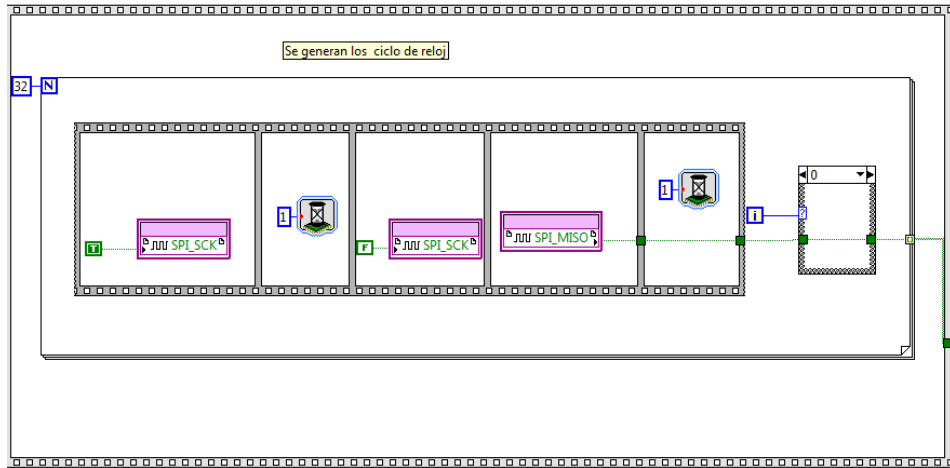


Figura 3.10 Adquisición voltaje analógica (parte II)

En el extremo derecho de la figura 3.11 se muestra la división de los dos canales analógicos y el acondicionamiento de los datos para representarlos digitalmente. La figura 3.12 muestra el SubVi del ADC para incorporarlo al programa general.

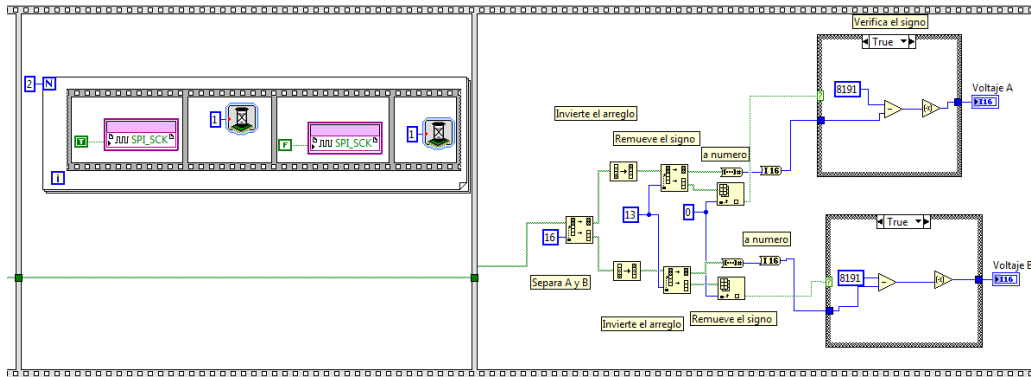


Figura 3.11 Adquisición voltaje analógico (parte III)

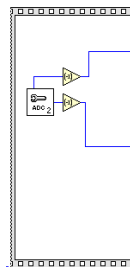


Figura 3.12 SubVI ADC

3.2.3 Programación del algoritmo INC

La figura 3.13 representa la figura 3.3 en Labview. Se ejecuta primero la estructura *Sequence* externa; es decir, se programa la ganancia del preamplificador (primer recuadro a la izquierda). Posteriormente se ejecuta el segundo recuadro de la estructura *Sequence*, se leen las entradas de los canales A y B (subVi etiquetado como ADC2); los valores se guardan en localidades de memoria y se repite la lectura un segundo más tarde. Los indicadores mostrarán el voltaje actual $V(i)$, la corriente actual $I(i)$, el voltaje guardado el segundo anterior $V(i-1)$ y la corriente anterior $I(i-1)$. Este proceso de lectura de voltaje se desarrolla continuamente por medio de la secuencia *while*, hasta que se detenga el programa manualmente.

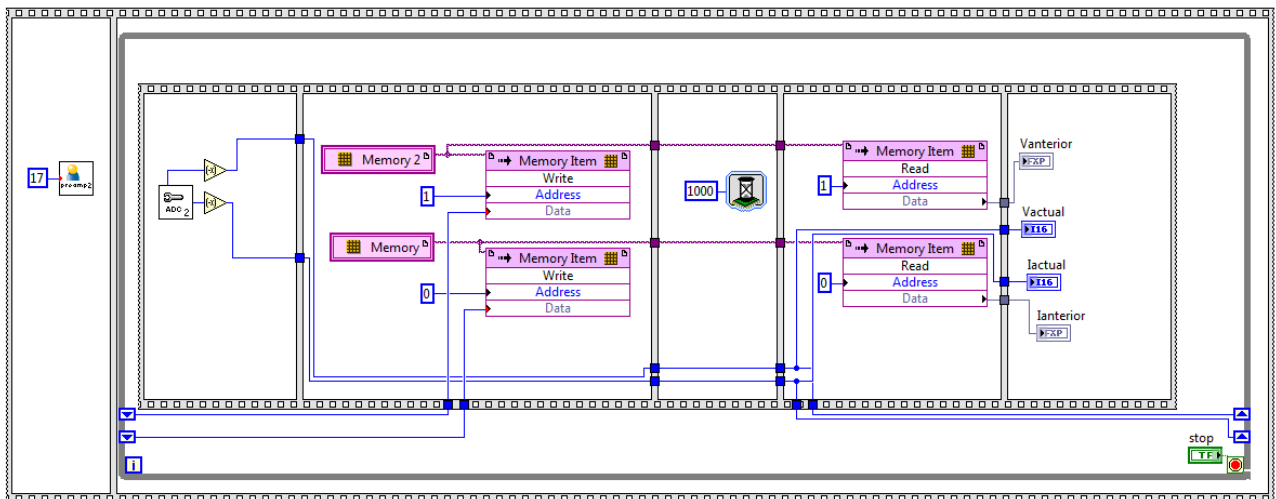


Figura 3.13 Programa INC para Adquisición y almacenamiento de voltajes y corrientes.

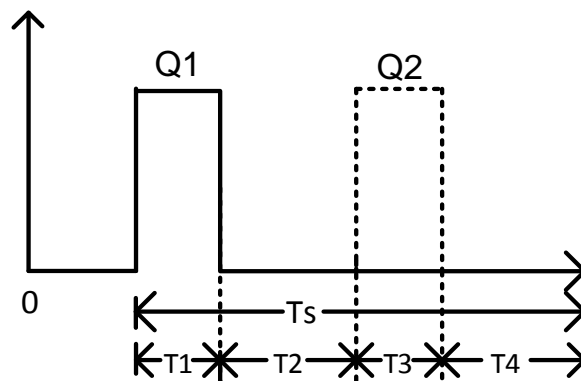


Figura 3.14 Señales de los MOSFET del Push-Pull

La figura 2.14 muestra las señales de conmutación de Q1 y Q2 para el push-pull. El funcionamiento es el siguiente:

- En T1 se envía un 1 lógico a la salida del FPGA etiquetada como FX2_I02_J1_1, y un 0 para la segunda salida FX2_I03_J1_2. Estas salidas se mantendrán durante el tiempo de encendido ton, el cual se obtendrá de multiplicar el periodo de conmutación por el ciclo de trabajo D.
- T2 muestra ahora el tiempo muerto de ambas señales, que será la diferencia del periodo total T_s menos T1. T4 será igual al resultado dividido entre dos.
- En T3 se invierten los estados de las salidas FX2_I03_J1_2 y FX2_I02_J1_1, permaneciendo durante un lapso igual a T1.
- El T4 es igual al T2, ambas señales permanecen apagadas.

Cabe mencionar que el bloque *wait* (representado como un reloj en Labview) tiene las siguientes unidades de tiempo: μseg , mseg y *tick*. El *tick* hace referencia al reloj del FPGA, para obtener una mayor resolución y poder variar el PWM con mayor precisión. La frecuencia del reloj es de 50Mhz, con un periodo de 20 nseg; por lo tanto, cada *tick* del *wait* equivale a 20 ns. A una frecuencia de 100 kHz le corresponde un tiempo de 10,000 nseg, la cual consume 500 *tick* [3]. La variable Data establece el control para cambiar el ciclo del PWM.

El programa de la figura 3.15 implementa la modulación PWM, conforme a la figura 3.14. Las dos entradas a este programa son el tiempo y el ciclo de trabajo D; sus salidas son las señales PWM. La variable local Data representa el ciclo de trabajo D guardado durante un segundo.

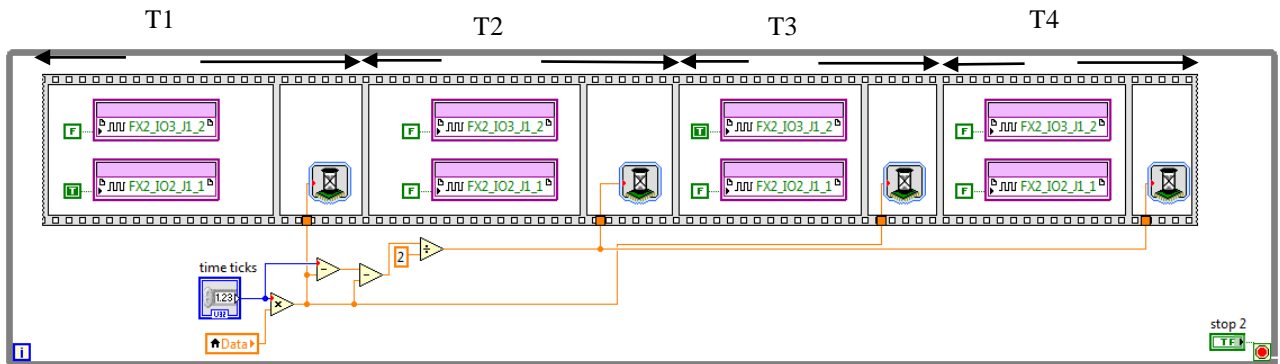


Figura 3.15 Programa del PWM

En la figura 3.16 se muestra el diagrama a bloques del programa para guardar el D durante un segundo y enviarlo posteriormente a la entrada del programa del PWM (figura 3.15), obteniéndose este valor de la salida del algoritmo INC. En la figura 3.17, la entrada al programa es D, se envía a una localidad de memoria, se guarda durante un segundo y se envía al programa de PWM como se muestra en la figura 3.16.

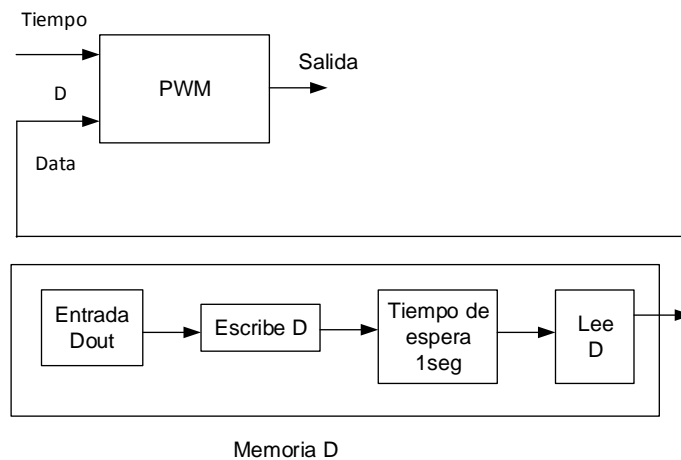


Figura 3.16 Programa de actualización del PWM

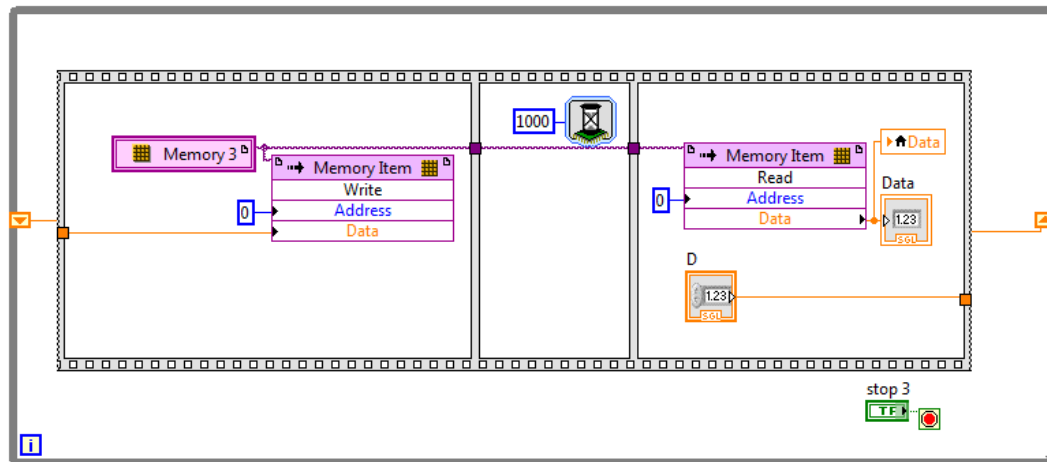


Figura 3.17 Memoria D

3.2.4 Programa Host para algoritmo INC

Debido a la capacidad de los módulos de memoria de Labview para guardar los datos analógicos y el ciclo de trabajo (figuras 3.13-3.15 y 3.17), al FPGA le quedan pocos recursos disponibles para llevar a cabo operaciones aritméticas, como se muestra en el anexo B. Además, como se explicó en la sección 3.11, no se cuenta con herramientas dentro del FPGA para programar el algoritmo INC. Se opta entonces por realizar las operaciones aritméticas y la programación INC en la computadora. Este método se denomina Host y tiene la peculiaridad de estar en el mismo proyecto y utilizar todas las herramientas de Labview sin ocupar la memoria del FPGA. Para mayor detalle acerca de la configuración del Host ver [4].

La figura 3.18 muestra el diagrama de bloques del Host. Las entradas analógicas se obtienen del programa del INC (figura 3.13). Estas entradas deben acondicionarse de acuerdo a la ecuación 2.4 (véase la figura 2.22) y a la ecuación (3.1).

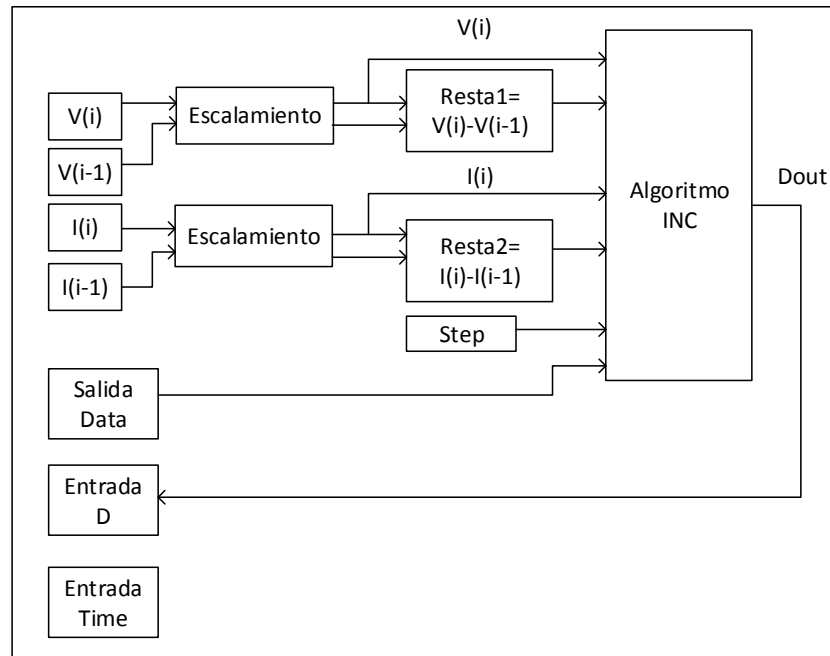


Figura 3.18 Diagrama de bloques del programa Host

$$y = mx + b \quad (3.1)$$

$$m = \frac{y_2 - y_1}{x_2 - x_1} = \frac{2.9 - 0.4}{8192 - (-8192)} = 152.5878 \mu \quad (3.2)$$

$$y = [(152.5878 \mu)(\text{salida del ADC})] + 1.65 \text{ V} \quad (3.3)$$

En el Host se implementa la ecuación 3.3 (figura 3.18) para mostrar el voltaje que se sensa a la entrada del ADC, en el recuadro etiquetado como “Escalamiento”. Posteriormente se calcula la diferencia entre el voltaje actual y el anterior como lo indica el algoritmo (ver figura 2.1 y figura 3.18). El recuadro de texto en la figura 3.19 (algoritmo INC) es el llamado nodo de formula. Tiene las siguientes entradas:

- Voltaje y corriente actuales $V(i)$ e $I(i)$
- “resta1” equivale a la diferencia del voltaje actual y el voltaje anterior

- “resta2” equivale a la diferencia de corriente actual y la corriente anterior
- Ciclo de trabajo D
- El paso *step* con el cual se modificará el ciclo de trabajo.

El algoritmo INC realiza las comparaciones necesarias (figura 2.1) para modificar el ciclo de trabajo. Se establece un intervalo de $\pm 0.01V$ como valor de 0 V, para acomodar el ruido en los canales analógicos.

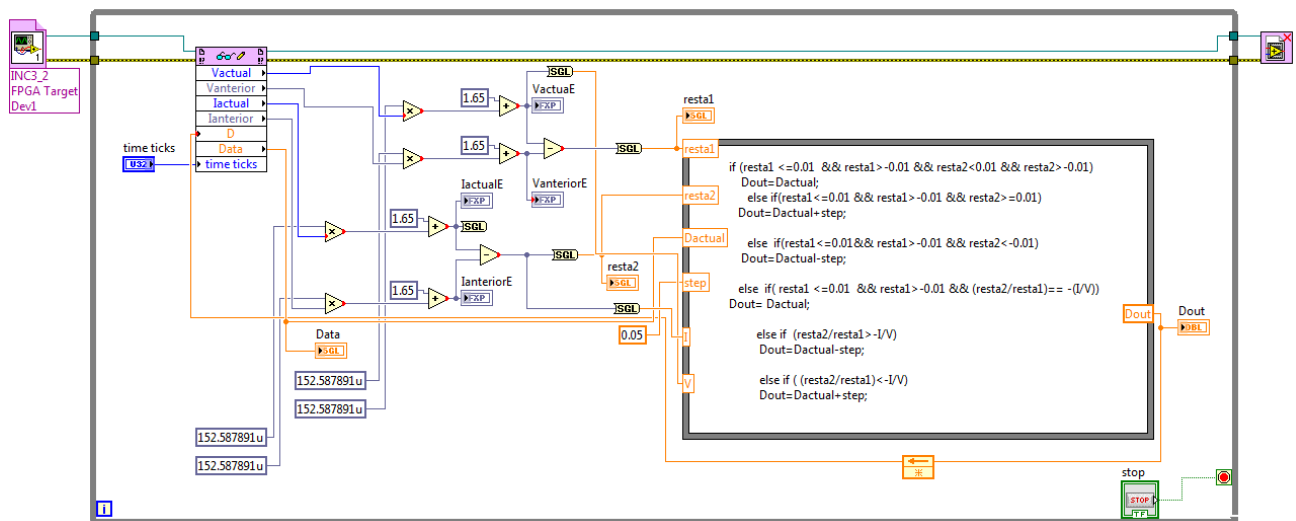


Figura 3.19 Programa Host del algoritmo INC

La figura 3.20 muestra el panel frontal del programa del Host del INC. El tiempo al que se trabajará es a $10 \mu s$ para el convertidor Push-Pull.

time ticks	Data	VactuaE	VanteriorE	resta1
0	0	0	0	0
	Dout	IactuaE	IanteriorE	resta2
	0	0	0	0

Figura 3.20 Panel frontal del Host

3.3 Bibliografía:

- [1] N. Instruments, LabVIEW FPGA, En línea. Fecha de acceso: 11 de junio del 2015. Disponible en : http://www.bo.infn.it/~falchier/teaching/Labview_fpga.pdf
- [2] N. Instruments, “*Using a Host VI to Communicate with the FPGA Target*”, En línea. Fecha de acceso: 11 de junio del 2015. Disponible en: http://zone.ni.com/reference/en-XX/help/371599H-01/lvfpgaconcepts/pfi_comm/
- [3] N. Kehtarnavaz y S. Mahotra, “*Digital signal Processing Laboratory Labview-Based FPGA implementation*”, editorial BrownWalker, pp 117-123, 329, 2010.
- [4] Getting Started with the LabVIEW FPGA Module and the Spartan-3E Starter Board, En línea. Fecha de acceso: 18 de junio del 2015. Disponible en: <https://wiki.gsi.de/pub/NIUser/SpartanStarterKitFPGA/Spartan3EStarterKitFPGAGettingStarted.pdf>

Resultados

En este capítulo se realizan las pruebas para comprobar el comportamiento del algoritmo INC, es decir los cuatro casos de éste como lo muestra la figura 2.1. Para ello no es necesario contar con el emulador FV y el convertidor Push-pull, debido a que se introducen los voltajes directamente al ADC del FPGA. Mediante el uso de dos fuentes de voltaje analógicas y divisores de tensión, se envían los voltajes a los dos canales analógicos del FPGA y se varían dependiendo, cual rama del algoritmo cuyo funcionamiento se desee comprobar. Esto dará como resultado en un aumento o disminución del ciclo de trabajo. Se muestran los voltajes desplegados en el panel frontal de Labview, y las formas de onda que se capturaron en el osciloscopio. A fin de apreciar visualmente los cambios en el ciclo de trabajo, se elige un paso de 0.05, aunque el paso puede ser de 0.01 e incluso de 0.001.

4.1 Caso 1: izquierda del MPP

$$\Delta V \neq 0, \quad \frac{\Delta I}{\Delta V} > -\frac{I}{V}, \quad D(i+1) = D(i) - \Delta D \quad (4.1)$$

- a) Condición inicial: En el panel frontal de la figura 4.1 se muestra un $D = 0.3$. La señal del osciloscopio (Fig. 4.2) muestra el ancho del pulso con los marcadores a y b ubicados para observar de manera más clara el cambio en D .

time ticks	Data	VactualE	VanteriorE	resta1	ΔV
441	0.3	0.528595	0.528564	0	
	Dout	IactualE	IanteriorE	resta2	ΔI
	0.3	1.38858	1.38306	0.00549316	

Figura 4.1 Panel frontal $D=0.3$

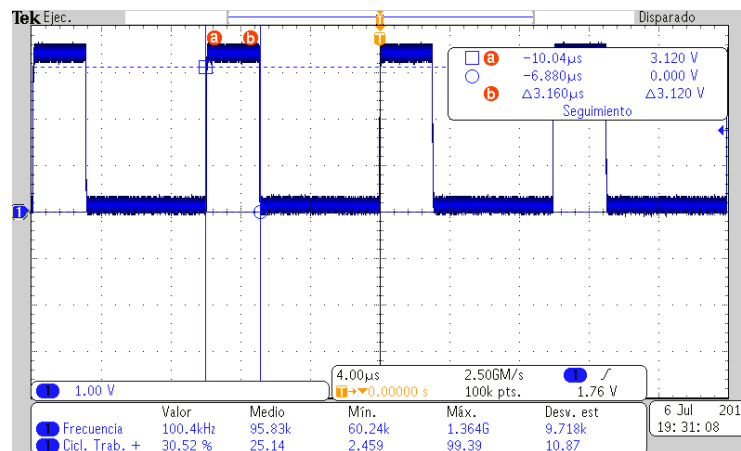


Figura 4.2 Señal de osciloscopio a $D=0.3$

- b) Cambios en las entradas analógicas: La figura 4.3 muestra las variaciones de voltaje y de corriente. Se ejecuta la operación de (4.1). El indicador del panel frontal “resta1”

corresponde a ΔV , “resta2” corresponde a ΔI , I corresponde a la corriente actual y V al voltaje actual. Sustituyendo los valores de voltaje y corriente en (4.2):

$$\frac{\Delta I}{\Delta V} = \frac{\text{resta2}}{\text{resta1}} = \frac{-0.0194092}{0.0078125} = -2.484 \quad (4.2)$$

$$\frac{-I}{V} = \frac{-1.36462}{0.53653} = -2.543 \quad (4.3)$$

Como se observa en (4.2) y (4.3) el cociente $\text{resta2}/\text{resta1}$ es mayor a $-I/V$. En consecuencia, D disminuirá en ΔD , como se muestra en el indicador D_{out} (figura 4.3).

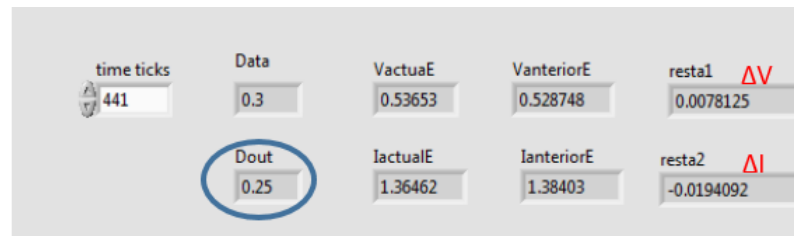


Figura 4.3 Disminuye D, $D_{out}=D-\text{step}=0.3-0.05=0.25$

- c) Corrección del ciclo de trabajo: En la figura 4.4 se muestra la disminución en D_{out} , el cual disminuyó 0.05 con respecto al valor actual. En el osciloscopio de la figura 4.5 se muestra claramente la disminución del ciclo notándose la diferencia en los indicadores a y b .

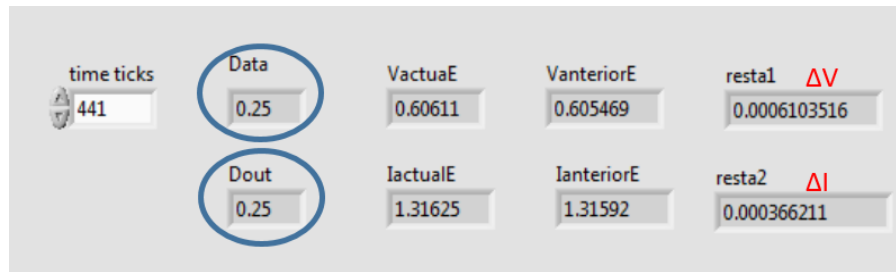


Figura 4.4 D=0.25 actualizado

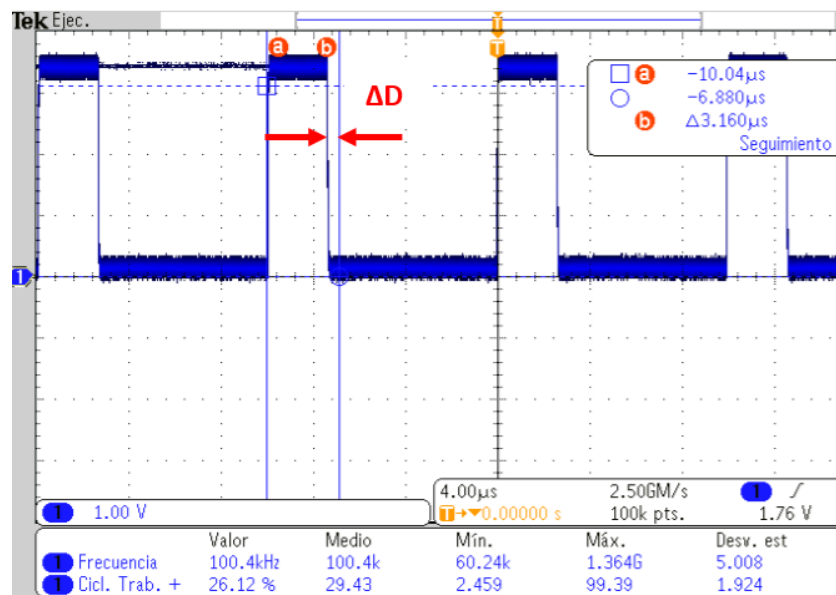


Figura 4.5 D=0.25, Dout= Data

4.2 Caso 2: derecha MPP

$$\Delta V \neq 0, \quad \frac{\Delta I}{\Delta V} < -\frac{I}{V}, \quad D(i+1) = D(i) + \Delta D \quad (4.4)$$

- a) Condición inicial: La figura 4.6 muestra un $D = 0.24$ tanto en el panel frontal como en el osciloscopio (figura 4.7); Los valores son los actuales.

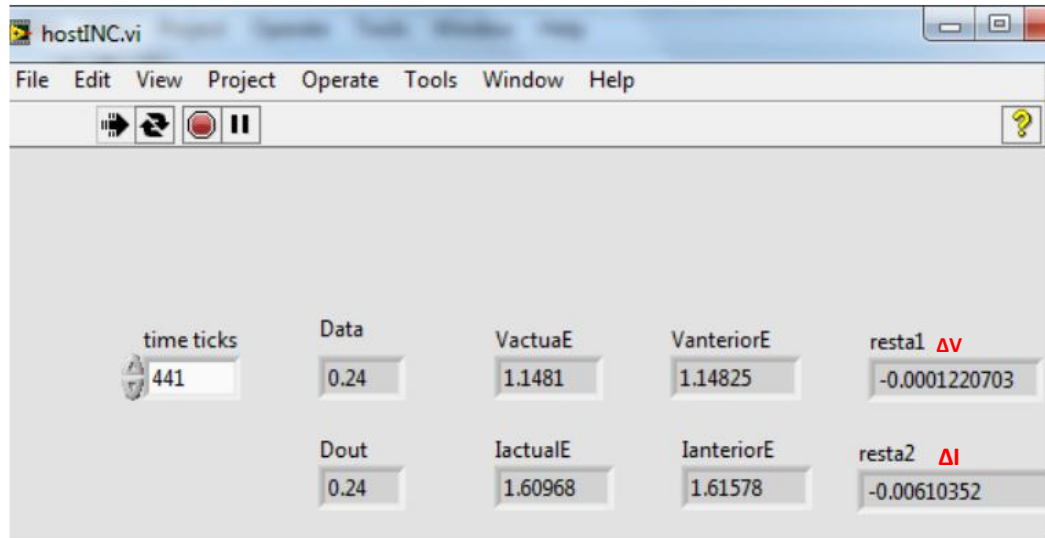


Figura 4.6 D=0.24 actual en panel frontal

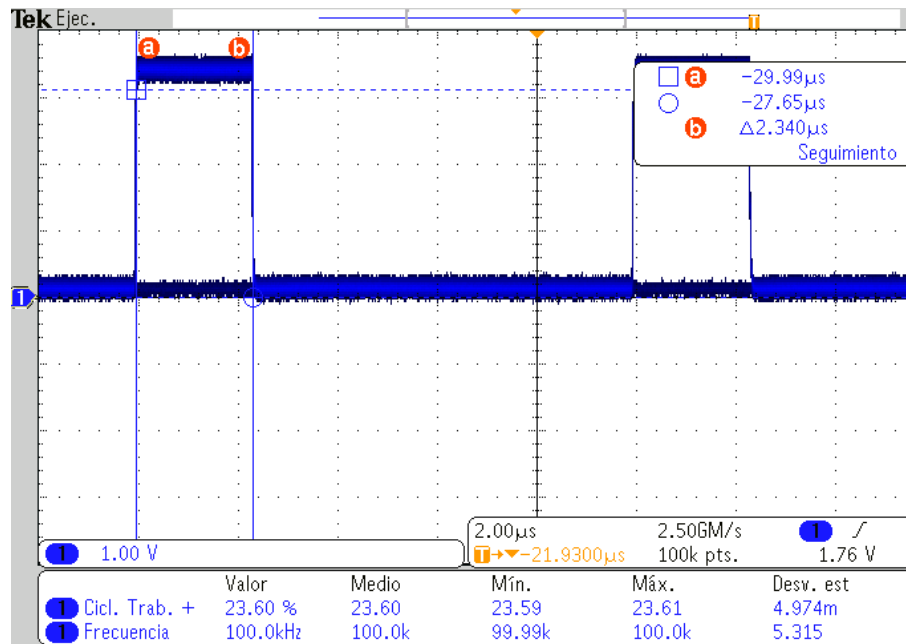


Figura 4.7 D=0.24 actual en osciloscopio

- b) Cambios en las entradas analógicas: Para comprobar el caso 2, se disminuye el voltaje (canal A) y la corriente se aumenta (canal B) como se muestra en la figura 4.8. Se sustituyen los valores en (4.4).

$$\frac{\Delta I}{\Delta V} = \frac{resta2}{resta1} = \frac{0.01928711}{-0.012085} = -1.5959 \quad (4.5)$$

$$\frac{-I}{V} = \frac{-1.64096}{1.12384} = -1.4601 \quad (4.6)$$

Como se muestra en (4.5) y (4.6), $resta2/resta1 < -I/V$, lo que constituye un aumento en el $Dout$ (figura 4.8).

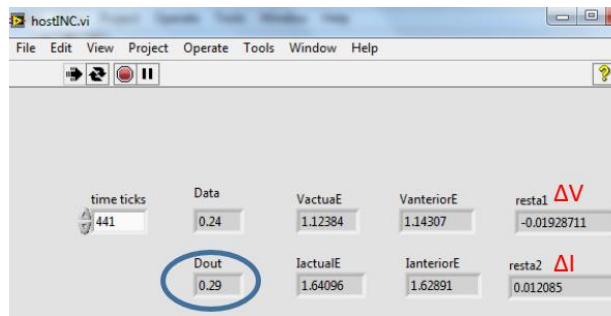


Figura 4.8 $Dout = Dactual + step (0.05)$

- c) Corrección del ciclo de trabajo: Después de un segundo el D se actualiza como se muestra en la figura 4.9, observándose el cambio en el osciloscopio (figura 4.10).

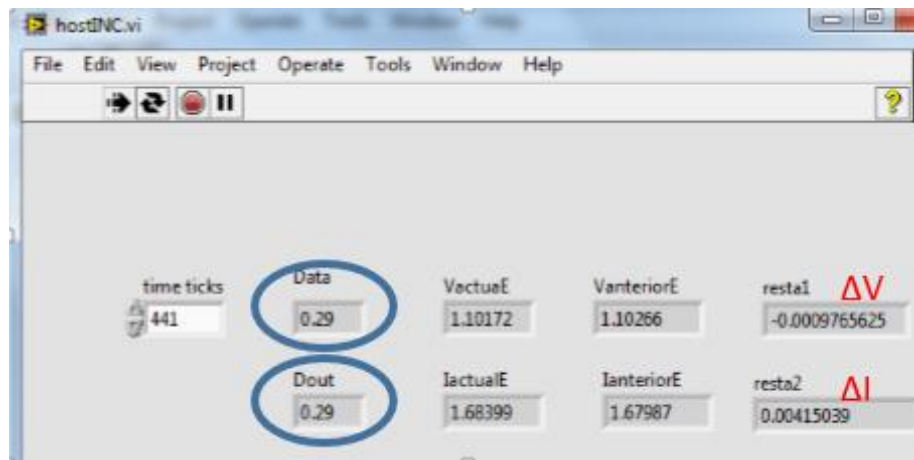


Figura 4.9 D=0.29 actualizado en panel frontal

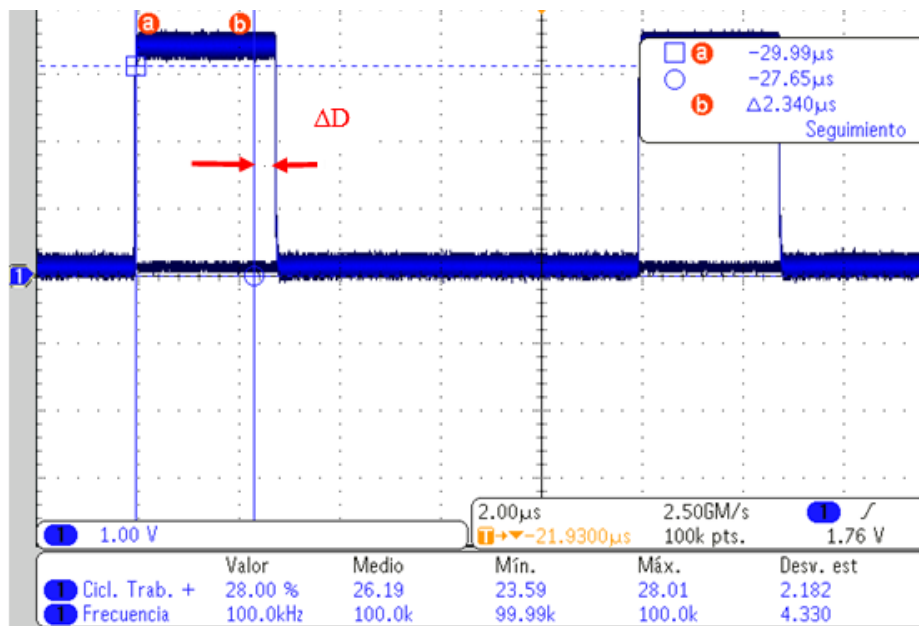


Figura 4.10 D=0.29 actualizado en osciloscopio

4.3 Caso 3: disminución de irradiancia

$$\Delta V=0, \quad \Delta I < 0, \quad D(i+1) = D(i) - \Delta D \quad (4.7)$$

- a) Condición inicial: En este caso, como lo indica (4.7), el voltaje permanece sin cambios. La variación de la corriente es menor a cero (es decir: la corriente actual desciende con respecto a la anterior), lo cual disminuye el D . En la figura 4.11 se muestran los voltajes, corrientes y D actuales, así como el PWM en el osciloscopio (figura 4.12).

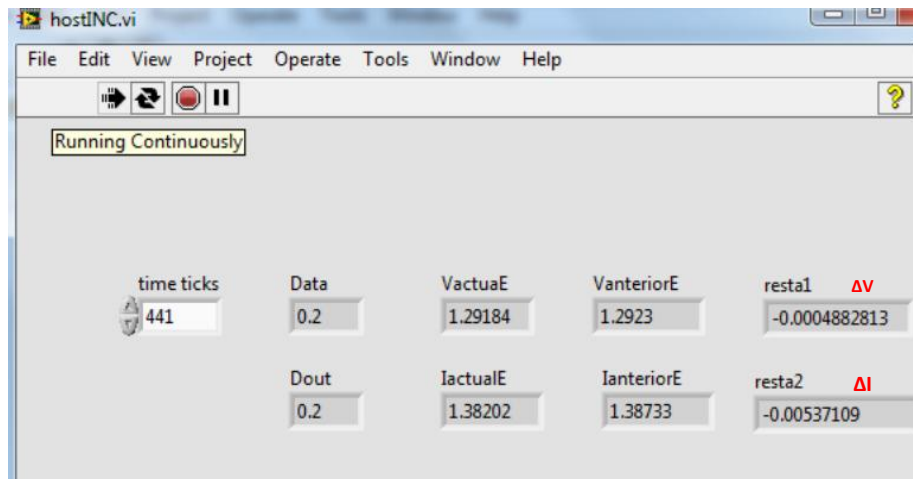


Figura 4.11 $D=0.20$ en panel frontal

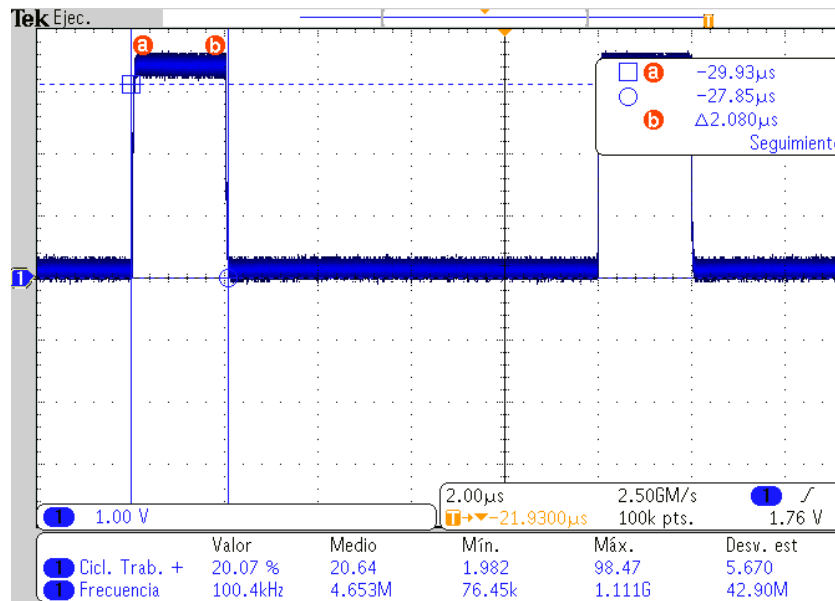


Figura 4.12 $D = 0.20$ en osciloscopio

- b) Cambios en las entradas analógicas: En el panel frontal de la figura 4.13 el voltaje permanece constante y disminuye la corriente; en el Dout se tiene $0.2 - 0.05 = 0.15$.

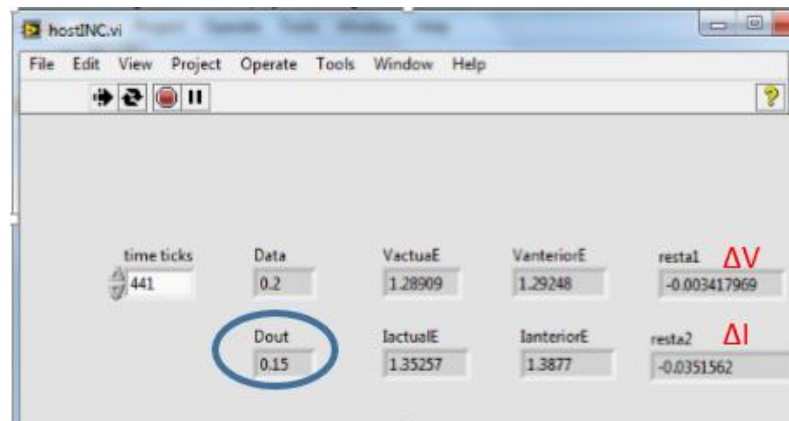


Figura 4.13 $D_{out} = D\text{-step} = 0.15$

- c) Corrección del ciclo de trabajo: la figura 4.14 muestra el valor de D actualizado en el indicador Data y en el osciloscopio (figura 4.15).

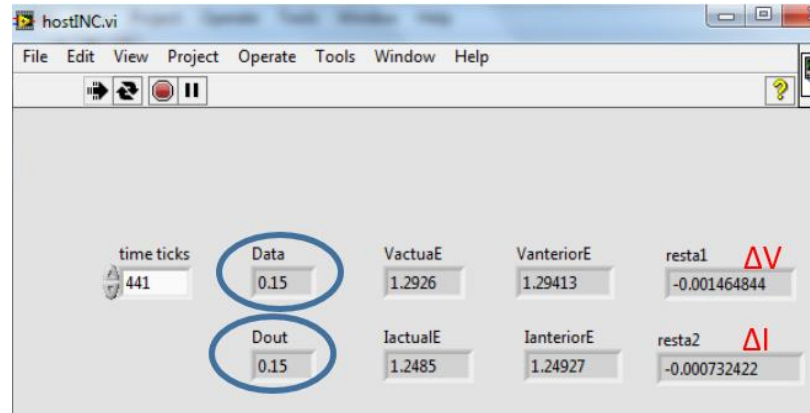


Figura 4.14 D=0.15 Actualizado

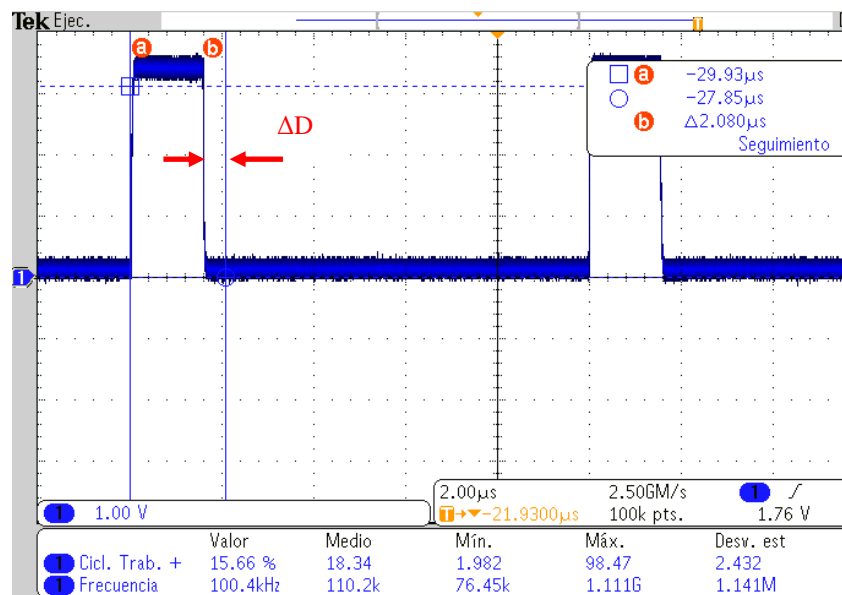


Figura 4.15 Data=Dactual Actualizado

4.4 Caso 4: aumento de irradiancia

$$\Delta V=0$$

$$\Delta I>0$$

$$D(i+1)= D(i)+ \Delta D$$

$$(4.8)$$

- a) Condición inicial: En este caso el voltaje se mantiene constante y se aumenta la corriente (canal B), lo cual traerá un aumento en el D . La figura 4.16 muestra un ciclo de 0.2 tanto en el indicador $Dout$ como en indicador $Data$. En el osciloscopio se puede apreciar este mismo valor (fig 4.17), cabe mencionar que aún no se ha variado la corriente.

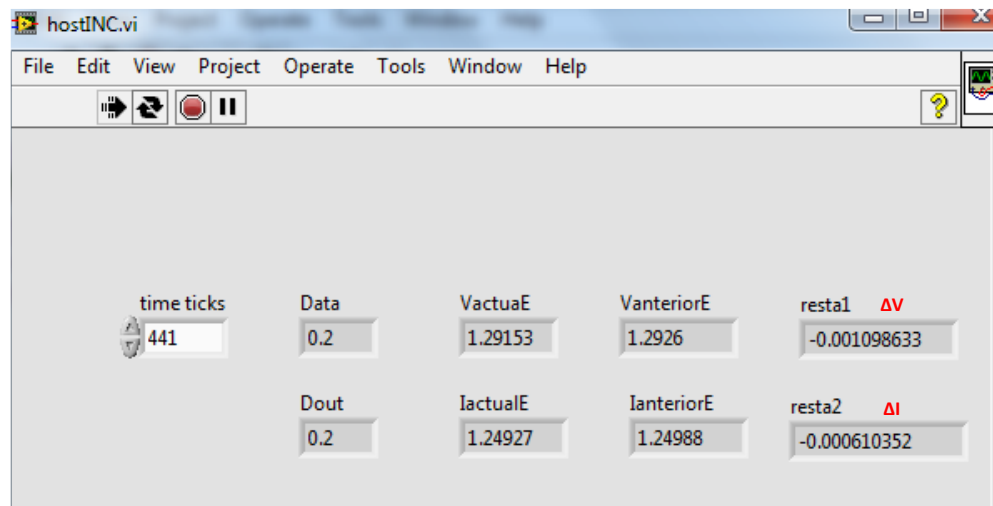


Figura 4.16 $D=0.20$, caso 4 en panel frontal

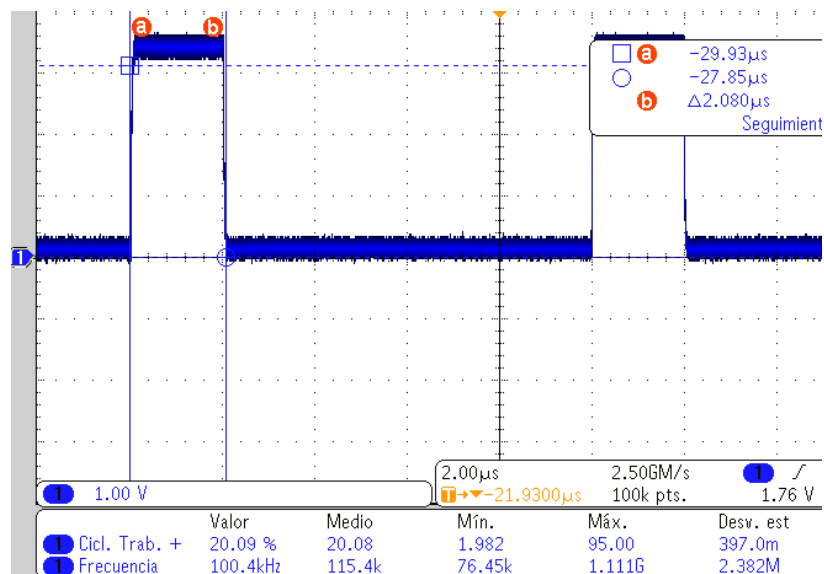


Figura 4.17. $D = 0.20$, caso 4 en osciloscopio

- b) Cambios en las entradas analógicas: En la figura 4.18 el voltaje (canal A) permanece constante y se aumenta la corriente, lo cual trae un aumento del ciclo como lo indica $D_{out} = D + \text{step}$; el resultado será $D_{out} = 0.2 + 0.05 = 0.25$, como se muestra en el panel frontal.

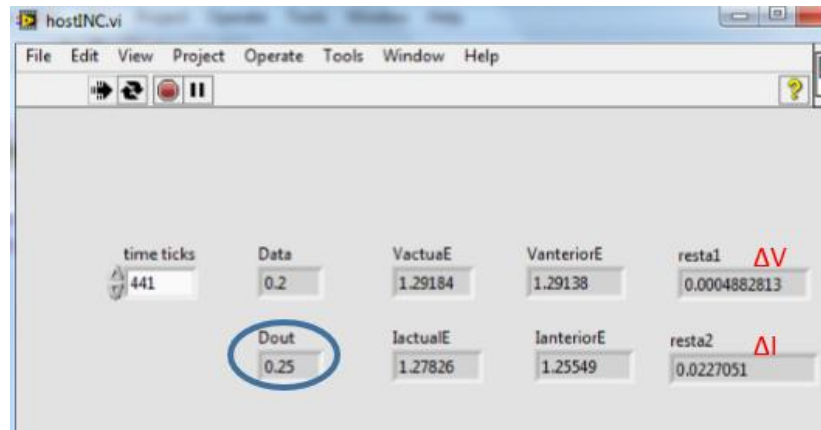


Figura 4.18 $D_{out} = D + 0.05 = 0.25$

- c) Corrección del ciclo de trabajo: Transcurrido un segundo se actualiza el ciclo de trabajo del PWM como se muestra en la figura 4.19 en el indicador de D_{out} y $Data$ del panel frontal, lo mismo que en el osciloscopio (figura 4.20).

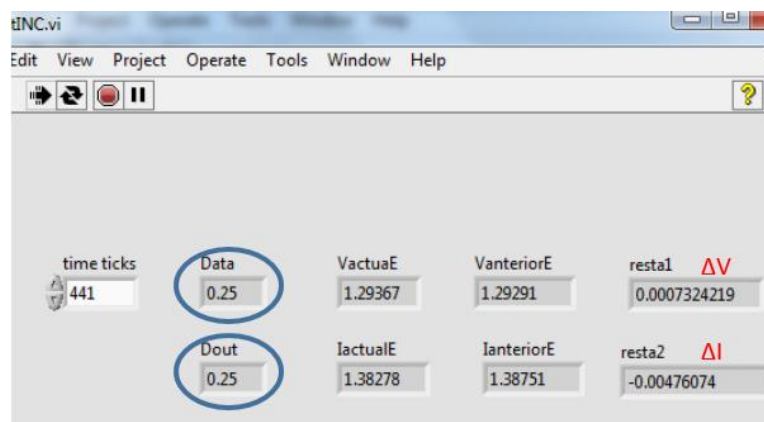


Figura 4.19. $D = 0.25$ actualizado en panel frontal

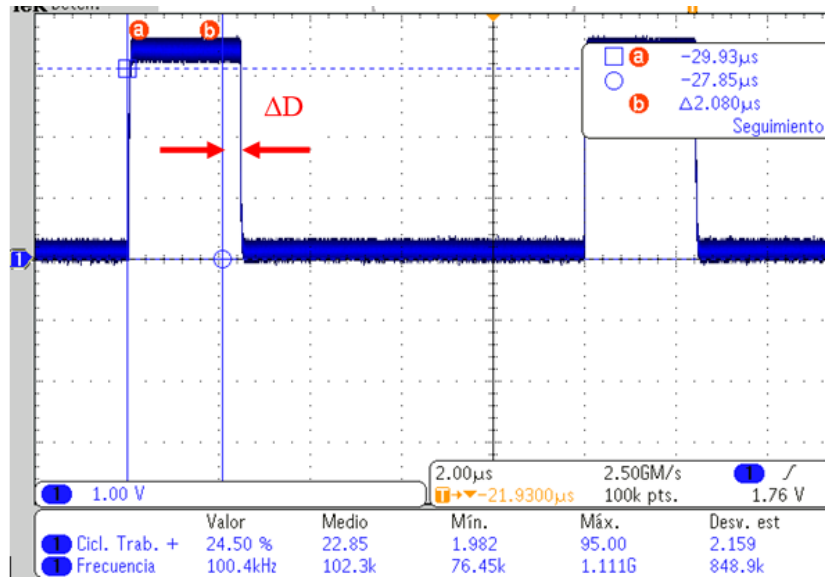


Figura 4.20. D = 0.25 actualizado en osciloscopio

Conclusiones

Con respecto a la literatura, se encontraron 13 técnicas para realizar el seguimiento del punto de máxima potencia. Los más populares son los algoritmos de P&O, INC y lógica difusa. Se decidió implementar el algoritmo INC por su complejidad media y por no presentar oscilaciones en régimen permanente.

Se implementó satisfactoriamente el algoritmo INC a paso fijo en el FPGA Spartan 3E. Esta tarjeta tiene la ventaja de trabajar con la plataforma de Labview; sin embargo, algunos módulos aritméticos y de almacenamiento de datos consumen bastante memoria, por lo que se aprovecha el método Host para solucionar este problema. Además, ciertas funciones no se pueden programar dentro del FPGA. También hay que considerar que la programación está limitada a los bloques del Labview, siendo el diseñador parte importante en la programación.

Originalmente se planeó utilizar el algoritmo INC de seguimiento del MPP con paso de tamaño variable, con un punto de partida al 80% del V_{oc} (voltaje de circuito abierto), control directo e implementarlo en un FPGA. Sin embargo, debido al tiempo que tomó la implementación a paso fijo, la instalación del software para el Spartan 3E y el Spartan 6, además de mapear los elementos de acondicionamiento de señal y aislamiento para la tarjeta al convertidor, se determinó limitar la implementación al algoritmo de paso fijo.

En un principio se planeó utilizar la tarjeta SPARTAN 6; desafortunadamente, se tuvo la dificultad de que ésta no dispone de convertidores analógico-digital, lo que obligaría a desarrollar circuitos de acondicionamiento analógico, o bien, a comprar una tarjeta de adquisición de datos. Se decidió entonces utilizar la tarjeta Spartan 3E, ya que ésta sí dispone de canales analógicos de entrada, con la ventaja adicional de que es posible programarla utilizando módulos de Labview, lo que facilita la programación del algoritmo. Sin embargo, se tiene la desventaja de que Spartan 3E no cuenta con memoria suficiente para las demandas de ejecución del algoritmo INC.

El obstáculo se resolvió recurriendo al modo Host, como se menciona en la sección 3.1.1. La programación se lleva a cabo como si la totalidad del programa residiera en la tarjeta del FPGA, aunque en realidad una porción se ejecuta en una PC, en sincronía con la tarjeta.

Por otro lado, es difícil evaluar el seguimiento del punto de máxima potencia porque el emulador del que se dispone (una fuente marca Magnapower) es de potencia mucho mayor a la del prototipo. La fuente tiene una capacidad de 10 kW; cuando se utiliza a una potencia de 100 W se está utilizando al 1 % de su capacidad. Como se identificó en el trabajo de tesis de José Luis Silva [Capítulo 1 referencia 19], a estas escalas la respuesta de la fuente no es adecuada para el seguimiento del punto de máxima potencia.

En resumen, los resultados demuestran la viabilidad de implementar en un FPGA el algoritmo de conductancia incremental para el seguimiento del punto de máxima potencia. Como tarea futura está la implementación con un paso de tamaño variable. Para esto será necesario utilizar una tarjeta de mayor capacidad que la Spartan 3E, la cual deberá contar con canales analógicos de entrada. Un punto al cual se le dedica poca atención, pero que puede impactar de manera significativa en los tiempos de desarrollo es el relacionado con la compatibilidad adecuada de todos los elementos de programación involucrados.

Anexo A

Convertidor Push-Pull

A continuación se presenta el análisis del convertidor push pull para implementarlo en simulink de matlab.

Se analiza el convertidor push-pull (figura A.1) para el tiempo de conmutación T_s (figura A.2)

De acuerdo a las posiciones del interruptor (encendido, apagado y tiempo muerto) debido a que este convertidor opera en modo discontinuo) se tienen las siguientes configuraciones de circuitos resultantes (figuras A.3, A.4, A.5).

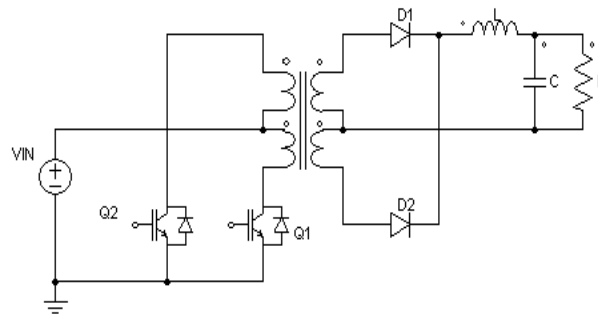


Figura A.1 Convertidor push-pull

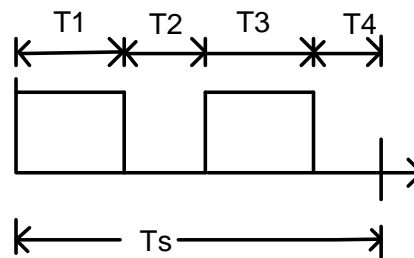


Figura A.2 Tiempos de conmutación de Q1 y Q2

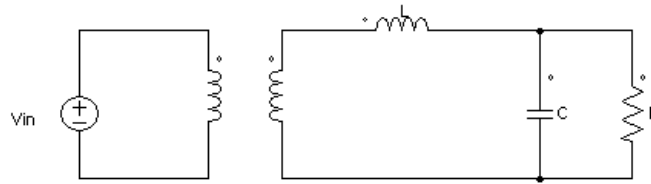


Figura A.3 Circuito equivalente 1 encendido

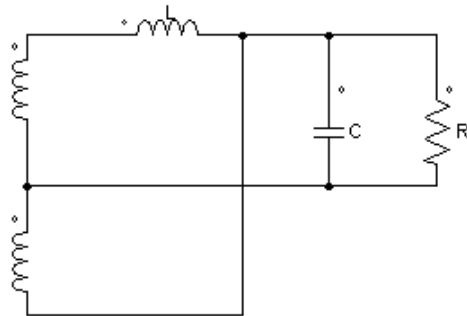


Figura A.4 Circuito equivalente 2 y 4, tiempo muerto

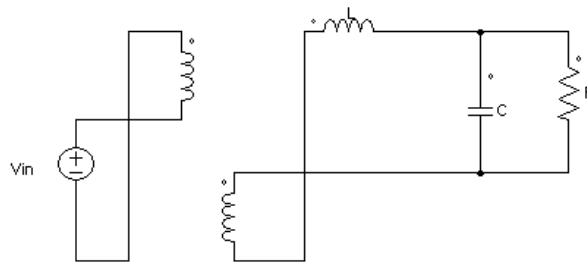


Figura A.5 Circuito equivalente 3

La figura A.1 muestra las ecuaciones diferenciales para el tiempo de encendido (t1 figura A.2)

son las siguientes:

$$V_{in} = L \frac{di_L}{dt} + V_c, \quad i_L = C \frac{dV_c}{dt} + \frac{V_c}{R} \quad (\text{A.1})$$

Reescribiendo las ecuaciones anteriores se tienen:

$$\frac{di_L}{dt} = \frac{1}{L}(V_{in} - V_c), \quad \frac{dV_c}{dt} = \frac{1}{C}\left(i_L - \frac{V_c}{R}\right) \quad (\text{A.2})$$

En el tiempo t2 se tienen las siguientes ecuaciones diferenciales

$$V_c = -L \frac{di_L}{dt}, \quad i_L = C \frac{dV_c}{dt} + \frac{V_c}{R} \quad (\text{A.3})$$

Reescribiendo las ecuaciones anteriores se tienen:

$$\frac{di_L}{dt} = -\frac{V_c}{L}, \quad \frac{dV_c}{dt} = \frac{1}{C}\left(i_L - \frac{V_c}{R}\right) \quad (\text{A.4})$$

Se observa que el tiempo de conmutación del convertidor push-pull (figura A.2) es dos veces al de un convertidor buck. Por lo cual las ecuaciones se multiplican por dos y por la relación de número de vueltas del transformador (N_s/N_p). La frecuencia de conmutación será dos veces que la del buck. Por lo tanto, se obtienen las ecuaciones diferenciales (A.5) y se simulan en la herramienta de Simulink de MATLAB.

$$V_c = 2 \frac{N_s}{N_p} u V_{in} - L \frac{di_L}{dt}, \quad i_L = C \frac{dV_c}{dt} + \frac{V_c}{R} \quad (\text{A.5})$$

Donde u es la señal PWM.

Anexo B

Memoria utilizada en FPGA

En la figura B.1 se muestra la compilación del programa del algoritmo INC en el FPGA mediante la plataforma de Labview. Se señalan en el recuadro rojo los porcentajes utilizados de los Slice LUTs, Slice register y bloques RAMs. Como se puede observar los porcentajes son algo considerables, sobre todo en el caso de los LUTs. Cabe mencionar que en Labview no se muestran los detalles de los elementos en cuanto al consumo de memoria del FPGA, sino hasta que se compila el programa se puede comprobar esto.

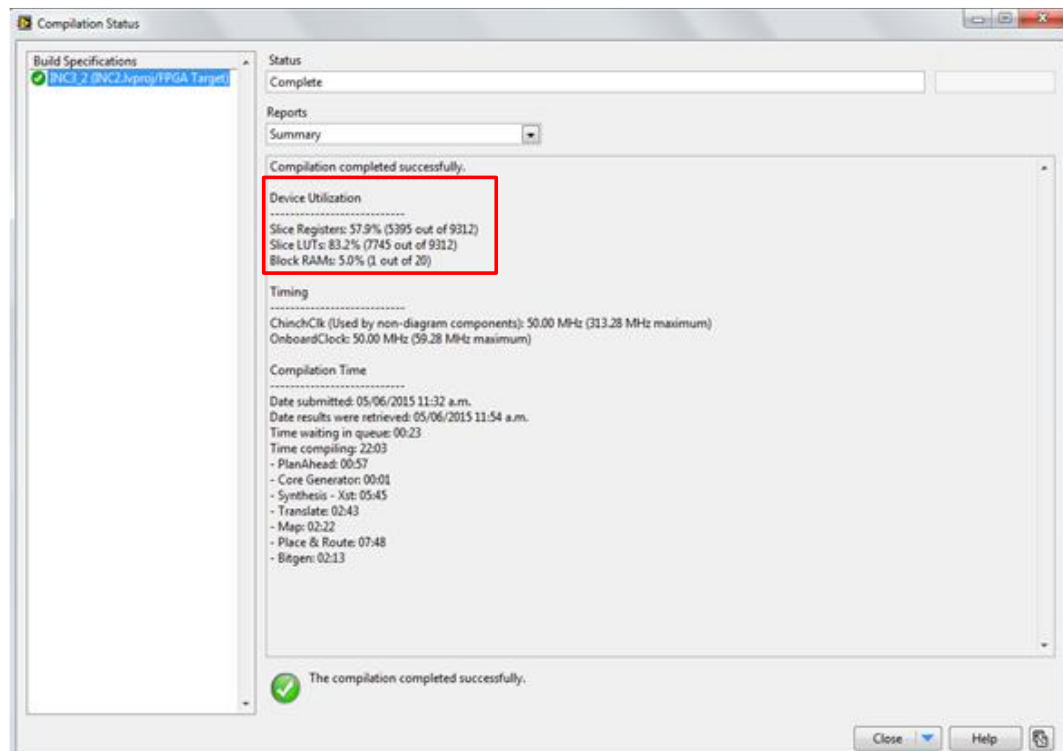


Figura B.1 Memoria utilizada del FPGA