

INSTRUMENTACIÓN VIRTUAL DE UN SISTEMA DE GENERACIÓN EOLOELÉCTRICO INTERCONECTADO A LA RED

Adolfo Rafael López Núñez

Tecnológico Nacional de México/Centro Nacional de Investigación y Desarrollo Tecnológico
adolfo_rafael@cenidet.edu.mx

Jesús Darío Mina Antonio

Tecnológico Nacional de México/Centro Nacional de Investigación y Desarrollo Tecnológico
jmina@cenidet.edu.mx

Roberto Carlos Gómez Hernández

Tecnológico Nacional de México/Centro Nacional de Investigación y Desarrollo Tecnológico
roca@cenidet.edu.mx

Gabriel Calderón Zavala

Tecnológico Nacional de México/Centro Nacional de Investigación y Desarrollo Tecnológico
gcalderon@cenidet.edu.mx

Oscar Hernández Martínez

Tecnológico Nacional de México/Centro Nacional de Investigación y Desarrollo Tecnológico
ohernandez@cenidet.edu.mx

Resumen

En este artículo se muestra el diseño e implementación de un instrumento virtual, el cual permite medir voltajes y corrientes de un generador eoloeléctrico interconectado a red, desarrollado como prototipo para pruebas de laboratorio. Dicho generador eoloeléctrico está basado en un generador doblemente alimentado, un convertidor back to back y su correspondiente sistema de control. El instrumento virtual se implementó en el software LabVIEW, utilizando como tarjeta de adquisición de datos la tarjeta myRIO. Además de adquirir y mostrar las

señales de voltaje y corriente en tiempo real del sistema eoloeléctrico, el instrumento virtual es capaz de almacenar los valores de dichas señales en un archivo para su posterior análisis. Cabe mencionar que este trabajo es la primera etapa de un trabajo más extenso, en el cual en un futuro se pretende ampliar las capacidades de la tarjeta myRIO, para adicionalmente implementar y monitorear estrategias de control del generador eoloeléctrico.

Palabras Claves: Instrumentación virtual, LabVIEW, myRIO, WECS.

Abstract

This article shows the design and implementation of a virtual instrument, which allows the measurement of voltages and currents of a grid connected Wind Energy Conversion System (WECS), developed as a prototype for laboratory tests. This WECS is based on a doubly fed generator, a back to back converter and its corresponding control system. The virtual instrument was implemented by using LabVIEW software, and the myRIO hardware as the data acquisition system. In addition to acquiring and displaying voltage and current signals in real time of the WECS, the virtual instrument is capable of storing the values of these signals in a file for further analysis. It is important to mention that this work is the first stage of a more extensive one; which is sought to expand the capabilities of the myRIO hardware, in order to additionally implement and monitor control strategies in the WECS.

Keywords: LabVIEW, myRIO, virtual Instrumentation, WECS.

1. Introducción

Actualmente el uso de energías renovables está cobrando importancia en el ámbito de la investigación científica, esto debido a las diversas ventajas y beneficios que ofrecen este tipo de tecnologías a la sociedad. Dentro de las energías renovables, un esfuerzo importante está orientado en la generación de energía eléctrica por medio de la energía eólica, la cual es hoy en día la de mayor penetración en el mercado de las energías renovables, con tasas de crecimiento anual superiores al 30% de acuerdo a [Sawin, 2016].

Gracias al avance de la electrónica de potencia, ha sido posible desarrollar Sistemas de Conversión de Energía Eólica (Wind Energy Conversion System: WECS) eficientes, de bajo costo y con flexibilidad para su interconexión con la red eléctrica. Una de las configuraciones en torno a los WECS se basa en un DFIG y un convertidor back to back, en donde la interconexión a red se hace en un punto de acoplamiento común (PCC por sus siglas en inglés). Dicho convertidor se encuentra constituido por un convertidor del lado del rotor (RSC, por sus siglas en inglés), un convertidor del lado de la red (GSC, por sus siglas en inglés) y por un capacitor de desacoplo entre ambos convertidores, figura 1.

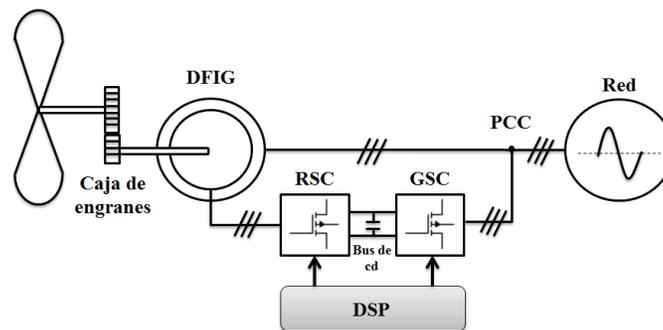


Figura 1 Diagrama esquemático del WECS basado en DFIG.

El convertidor back-to-back debe cumplir con los objetivos de control típicos de un sistema interconectado, en donde el RSC controla la transferencia de la potencia activa y reactiva entre el estator del DFIG y la red. Mientras que el GSC controla la potencia reactiva entre el convertidor y la red, además de encargarse de la regulación del bus de cd [Aguilar, 2015], [Jiménez, 2012].

En el laboratorio de electrónica del Centro Nacional de Investigación y Desarrollo Tecnológico (CENIDET) se cuenta con un WECS con las características previamente descritas; en donde se resalta que el control, figura 1, se ha implementado en un procesador digital de señales (DSP). Todo el sistema de control de este WECS, controladores del RSC y del GSC, está basado en control vectorial, su desarrollo se describe ampliamente en [Calderón, 2014].

Dado que se pretende que este WECS se constituya en un prototipo de laboratorio sobre el cual se puedan desarrollar diversas pruebas, como son sistemas de

monitoreo, pruebas de desempeño de diferentes estrategias de control, etc., es indispensable contar con una interface que permita la adquisición de las variables eléctricas más relevantes del sistema de una manera sencilla y en tiempo real a través de una computadora, dichas variables se muestran en la figura 2; así como también, que permita actuar desde la computadora para la modificación, entre otros, de las referencias de potencia, de las condiciones de operación del WECS (velocidad rotacional o del viento), etc.

De manera específica, el monitoreo de las variables mostradas en la figura 2 es de suma importancia para verificar el correcto funcionamiento del control implementado en los convertidores que conforman el back to back; en el caso del RSC, su control se realiza mediante la modificación de las corrientes del rotor, por lo que la visualización y registro de dichas corrientes, así como los voltajes y corrientes del estator queda justificado. Con respecto al GSC, el sistema desarrollado permitirá verificar la correcta regulación del bus de cd principalmente.

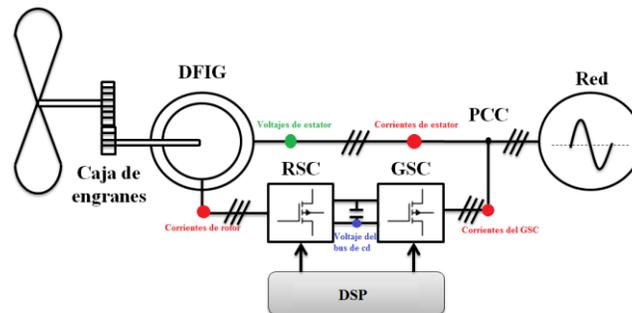


Figura 2 Puntos de sensado del WECS.

Cuando se trata de monitorear variables en un sistema, una opción es el uso de interfaces gráficas de usuario (GUI, por sus siglas en inglés). En el caso de WECS, en [Ayaz, 2016] los autores proponen el uso de una GUI basada en MatLab, para evaluar diferentes tipos de generadores para aplicaciones eólicas, específicamente, en términos de sus efectos sobre la red eléctrica. Se resalta que estos trabajos son GUIs de sistemas que son simulados y no interactúan con ningún subsistema real del WECS.

Por otro lado, en [Rajarajan, 2013] se presenta un esquema de simulación en tiempo real de un sistema de generación eoloelectrónica. El esquema propone la

integración de un modelo y controles del WECS, implementados en Simulink y una interface diseñada con LabView para el monitoreo del WECS. Para lograr una simulación en tiempo real se hace uso de la herramienta MatLab Real Time Workshop y del Simulation Interface ToolKit (SIT) de LabView. El sistema es capaz de intercambiar información entre el sistema que corre en Simulink y los Instrumentos Virtuales para el monitoreo implementados en LabView. En este trabajo, nuevamente el desarrollo es netamente basado en simulación y la interacción con la realidad es solo para fines de monitoreo.

Por su parte, [Topor, 2015] propone la emulación de una turbina eólica en donde el modelo de turbina eólica y el controlador del motor que emula a la turbina se implementa en el controlador en tiempo real basado en FPGA: *NI RIO 9068*. Este desarrollo cae dentro del concepto HIL (Hardware In the Loop). La ventaja de esta implementación es que a través de la NI RIO 9068 es posible usar un modelo virtual de una planta, en este caso de la turbina eólica.

Entre estas opciones, ninguna contempla la interacción con un WECS real, específicamente con el sistema de control y las variables asociadas a dichos controles; lo cual representa un reto diferente. Dentro de las opciones disponibles para la adquisición, monitoreo y almacenamiento de las variables sensadas, el uso de la instrumentación virtual ha tenido auge en los últimos años, esto debido a la conveniencia del usuario para definir la funcionalidad y apariencia de los instrumentos de medición según la aplicación que se trate. Diversos trabajos han enfocado sus esfuerzos en realizar la instrumentación virtual de sus sistemas o procesos por medio del software Labview [Karhe, 2013], [Osorio, 2010], [Saa, 2007]; en dichos trabajos, la presentación y almacenamiento de las variables sensadas se realizan a través de una computadora, sin embargo, la adquisición de estos datos generalmente se realiza a través de una interface diseñada de manera personalizada y basada en algún microcontrolador.

En esta aplicación en particular, el uso de un microcontrolador resultaría inapropiado, esto debido principalmente a que para tomar una muestra por cada variable es necesario una cierta cantidad de ciclos de reloj (dependiendo del microcontrolador a utilizar), por lo que la medición no se realizaría en tiempo real.

Lo anterior es de suma importancia, ya que en un futuro se pretende ampliar las capacidades del instrumento virtual, para de manera adicional, implementar y monitorear las estrategias de control del convertidor back to back.

Debido a lo anterior, para la implementación del sistema de instrumentación virtual de este trabajo, se consideró como opción más favorable el software de National Instruments LabVIEW en conjunto con la tarjeta myRIO, con la cual se realiza el procesamiento en tiempo real de las variables del caso de estudio.

2. Métodos

Como se mencionó previamente, para la implementación del instrumento virtual del WECS se utilizó el hardware de National Instruments myRIO como tarjeta de adquisición de datos, el nombre proviene de “Reconfigurable Input/Output” y es un dispositivo de hardware embebido que contiene un microprocesador doble núcleo *ARM Cortex-A9* a 667 MHz y un FPGA *Xilinx Zynq 7010* con 28,000 celdas lógicas programables, los cuales pueden ser programados de manera independiente. Además contar con velocidad máxima de muestreo de 500 kilo-muestras por segundo (kS/s), 12 bits de resolución entre otras características [National Instruments, 2013].

La programación de la tarjeta myRIO se llevó a cabo por medio del software LabVIEW a través de un paquete adicional denominado Software Bundle.

El Instrumento Virtual (VI, por sus siglas en inglés) se diseñó tomando en cuenta las siguientes consideraciones:

- **Adquisición de señales:** Se busca que el sistema de adquisición se implemente mediante la tarjeta myRIO.
- **Visualización de señales:** Se deben de mostrar en una pantalla los datos sensados en tiempo real.
- **Registro de datos:** Las señales de interés se deberán de almacenar en un archivo para su posterior análisis.

Para llevar a cabo las consideraciones anteriores se implementó la máquina de estados de la figura 3.

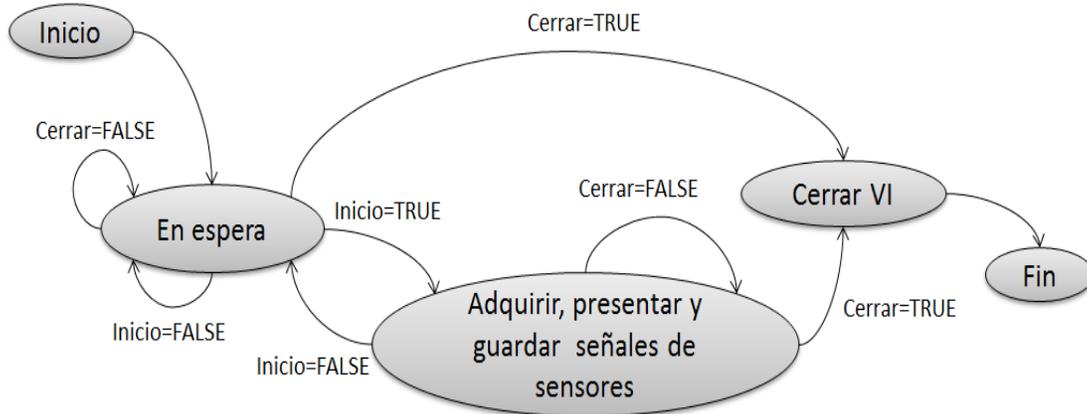


Figura 3 Diagrama esquemático del WECS basado en DFIG.

La descripción de los estados mostrados en la Ilustración anterior son los siguientes:

- **Inicio:** Utilizado para inicializar variables, limpiar gráficas y definir parámetros.
- **En espera:** La interfaz estando inactiva espera una acción del usuario como lo es activar los botones “Inicio” y “Cerrar” ubicados en el panel frontal del VI que se mostrará más adelante.
- **Adquisición/Presentación/Guardar:** La tarjeta myRIO realiza la adquisición de las señales y las entrega a la computadora para presentarlas en pantalla y almacenarlas en archivo.
- **Cerrar:** Se eliminan referencias creadas y se limpian variables para finalmente cerrar el programa.

La lógica de transición entre estados está definida por selectores lógicos, activados por los botones del Panel Frontal: Inicio, Cerrar y Guardar y que siguen la secuencia indicada en el diagrama de estados. Esto se encuentra dentro de una estructura case que contiene un caso por cada estado de la máquina, el cual se encuentra anidado en una estructura while para que sea un ciclo que sólo se detenga cuando el usuario cierre el programa, figura 4. En cada caso de la estructura case se puede colocar código en cada uno de los estados correspondiente a las acciones que se deseen realizar.

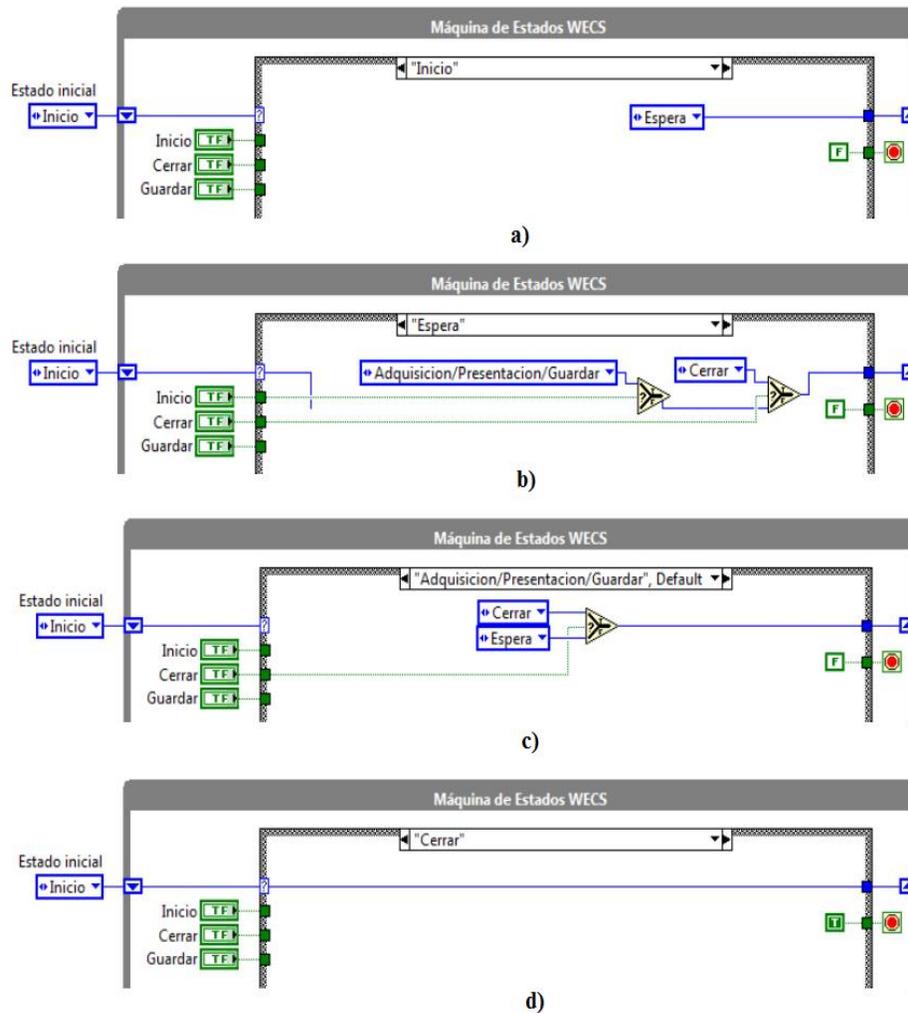


Figura 4 Diagrama de bloques de la máquina de estados.

Adquisición de Datos

El hardware de la tarjeta myRIO posee tanto un procesador como un FPGA, los cuales pueden ser programados de forma independiente. El uso del procesador de la tarjeta trae consigo los siguientes problemas:

- Los bloques de programación que permiten tomar una muestra por canal de la tarjeta necesitan una cantidad de ciclos de reloj para ejecutarse y no es constante. Esto deja una incertidumbre sobre la tasa de muestreo más adecuada para evitar muestreos en tiempos no constantes o, incluso, que el código utilice el cien por ciento del tiempo de procesador.
- LabVIEW maneja bloques ya preparados para almacenar los datos de adquisición, pero éstos necesitan estar en un tipo de dato llamado

Waveform que contiene un arreglo 1D de datos, un tiempo inicial t_0 y un periodo de muestreo dt . Los datos iniciales se almacenan en un tipo de dato flotante de 32 bits sin ningún atributo adicional y convertirlos hacia el tipo de dato Waveform resultó ser una operación costosa en procesamiento, limitando más la tasa de muestreo máxima de adquisición.

De modo que la acción más apropiada es realizar la adquisición con código alojado en la FPGA de la myRIO que se muestra en la figura 5.

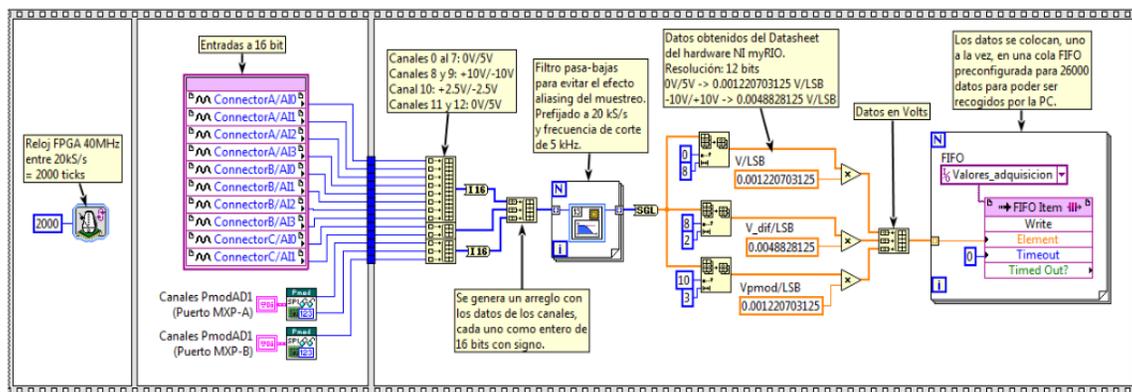


Figura 5 Diagrama de bloques de la adquisición de datos.

Esta acción resolvió los problemas principales, debido a que el FPGA presenta, por naturaleza, un comportamiento determinístico, comparado con el procesador. La descripción del funcionamiento de la figura 4 es la siguiente:

- El código del primer cuadro de dicha estructura corresponde a un bloque de tiempo de espera que asegura que el periodo de muestreo sea de dos mil *ticks* que son iguales a 50 microsegundos, una frecuencia de 20 kS/s.
- La adquisición de datos se efectúa mediante los canales disponibles de la tarjeta, para lo cual la tarjeta myRIO utiliza un ADC que muestrea cada canal y realiza la conversión con 12 bits de resolución. Estos datos son almacenados en valores de tipo entero de 16 bit que el programador puede utilizar en el código. Lo que se hace con ellos es guardarlos en un arreglo 1D (unidimensional) para manejarlos más cómodamente, usando el bloque Array Build.

- Posteriormente el arreglo de datos se hace pasar por un bloque de filtro Butterworth configurado como de tipo pasabajas, con una frecuencia de corte de 5 kHz, tasa de muestreo esperada de 20 kS/s y de segundo orden para evitar el efecto de Aliasing [Soria, 2003], es decir, que al reconstruir la señal muestreada se reproduzcan frecuencias incorrectas. El filtro sólo funciona con una muestra a la vez, de modo que en un ciclo for se hacen pasar una muestra a la vez para aplicar el procesamiento.
- Posteriormente se convierte el arreglo que presenta todos sus datos de tipo entero de 16 bits a datos de tipo flotante de 32 bits, después los datos se separan en sus canales correspondientes a sus fuentes de origen (FPGA I/O Node) utilizando el bloque Array Subset el cual extrae Y valores a partir de un índice X, guardándolos en otro arreglo más pequeño que el original.
- Habiendo separado los datos, se multiplican por el voltaje correspondiente al Bit menos significativo (LSB), basado en los voltajes límite que admite cada puerto. Después se separan los canales y con su valor correspondiente de Volts/LSB se multiplican y convierten de bits a Volts para finalmente agruparlos en un nuevo arreglo con el bloque Build Array.
- Finalmente se almacenan, uno a la vez, en una cola FIFO (primer dato que entra, primer dato que sale) de la FPGA, los elementos de esta cola FIFO pueden ser leídos desde el código del VI principal para su uso.

Registro/Almacenamiento de Datos

Para esta parte se creó un *subVI* que realiza la función de almacenamiento de las variables sensadas como valores separados por comas y almacenado en un archivo de texto plano con extensión *.csv.

Para crear el encabezado se escribió otro *subVI*, figura 6, el cual utiliza la fecha y hora del sistema que ejecuta el VI principal para generar el nombre de archivo donde se almacenarán los datos sensados, así como la ruta en disco donde se escribirá dicho archivo.

Este subVI se aloja en una estructura case para que se ejecute sólo cuando este se manda a llamar por primera vez, figura 7.

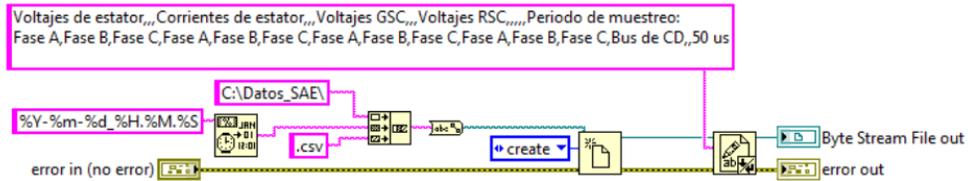


Figura 6 Código para crear el encabezado del archivo de valores separados por comas como un subVI.

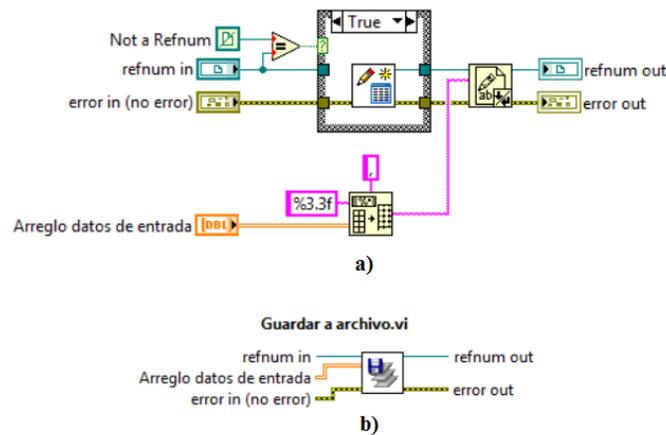


Figura 7 Código para el registro de los datos sensados como un subVI y icono del subVI para utilizarse en el VI principal.

Visualización de Datos

En lo que respecta a la visualización de los datos, estos se presentan en pantalla mediante un código alojado en el VI principal, específicamente en una estructura *while* mostrado en la figura 8.

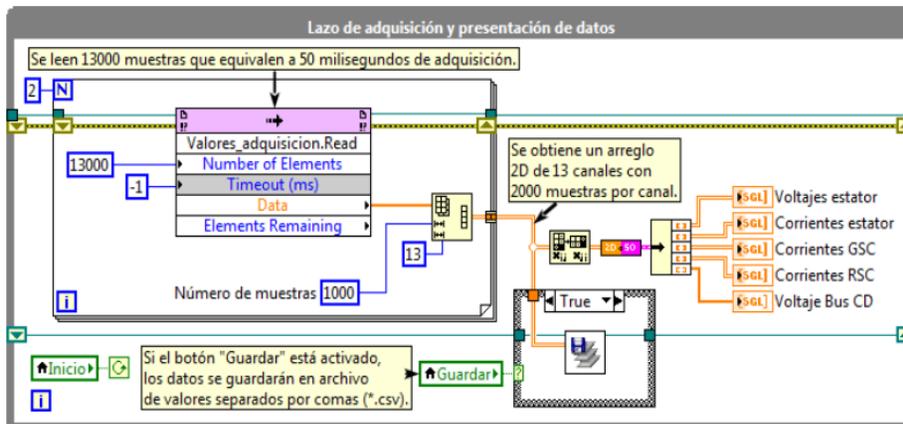


Figura 8 Estructura de adquisición y presentación de datos.

Primero se recogen los datos en la cola *FIFO* de la *FPGA* en una estructura *for*, posteriormente se convierte cada arreglo 1D de trece mil datos en un arreglo 2D de mil filas (número de muestras) por trece columnas (número de canales), las cuales se concatenen. El arreglo 2D resultante con un tamaño de dos mil muestras por trece canales equivalen a cien milisegundos de adquisición de modo que se cumple que para un segundo se presentarán diez de estos arreglos.

Para presentarlos en pantalla es necesario aplicar la transpuesta al arreglo 2D para luego pasar los datos a un *subVI*, el cual aplica un escalamiento a cada señal (Volts o Amperes según sea el caso) y junta las salidas de dichas señales para que puedan visualizarse en cinco gráficas, figura 9, las cuales mostrarán las señales del voltaje y corriente de estator, las corrientes del RSC y del GSC así como el voltaje del bus de cd.

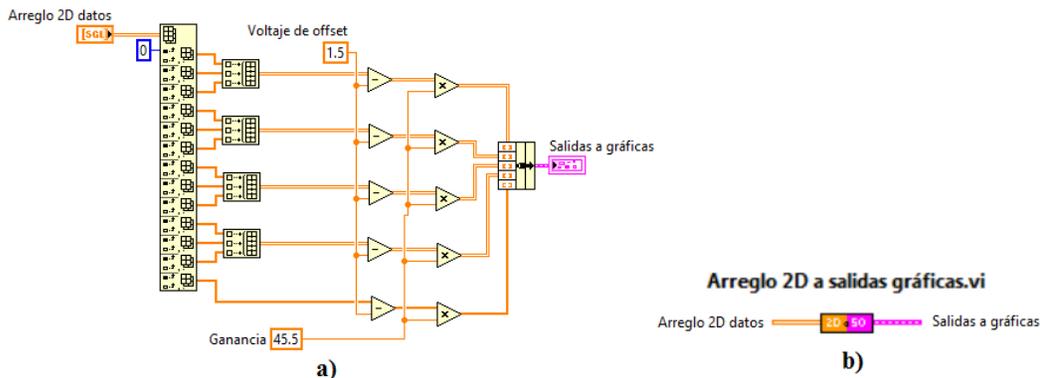


Figura 9 Código del subVI de conversión del arreglo 2D a cinco gráficas, ícono del subVI.

La salida de este *subVI* se vuelve a dividir usando la función *Unbundle* y cada salida se manda a una gráfica del *VI* principal, como se muestra en la figura 10.

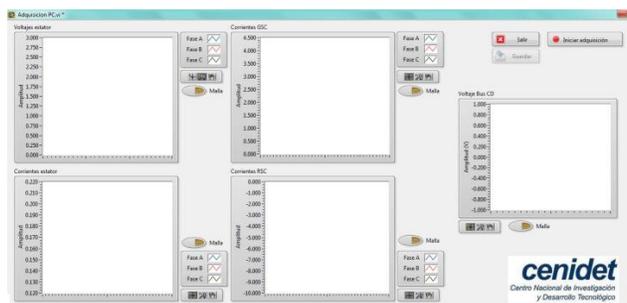


Figura 10 Panel frontal del instrumento virtual.

3. Resultados

Las pruebas realizadas tuvieron dos objetivos, verificar la correcta visualización de las variables medidas por el VI, además de verificar el correcto guardado de dichas variables. Debido a que el WECS ubicado en el laboratorio de electrónica no estaba operando en su totalidad al momento de realizar las pruebas del VI, únicamente se mostrarán los resultados correspondientes al voltaje de estator. La figura 11 muestra una de las pruebas realizadas en el laboratorio de electrónica.

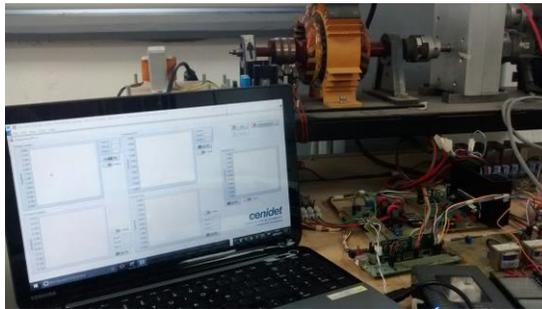


Figura 11 Adquisición de datos del WECS.

El voltaje de estator se obtuvo por medio de una tarjeta de sensado y de las cuales en este artículo no se reportará su diseño, dichas tarjetas entregan una salida de 0 a 3 V, que son el rango de voltaje que admite el DSP y están dentro del rango de voltaje analógico de entrada para la tarjeta myRIO (0 a 5 V); debido a que estas señales son senoidales, las tarjetas de sensado generan un voltaje de offset de 1.5 V.

En la figura 12 se muestra el correcto funcionamiento en la visualización de los voltajes de estator. Cabe mencionar que las gráficas que se observan en las variables que no sea el voltaje de estator no deben considerarse, pues es ruido ya que estas señales no se encontraban conectadas a la tarjeta myRIO debido a que el WECS no se encontraba trabajando en su totalidad.

Para la prueba de guardado en archivo se comprobó que el VI fue capaz de almacenar los datos medidos en la ruta "C:\Datos_SAE" como archivos con extensión *.csv, figura 13, los cuales se pueden abrir en un visor de archivos de texto o en una aplicación de hojas de cálculo como Microsoft Office Excel, figura 14.

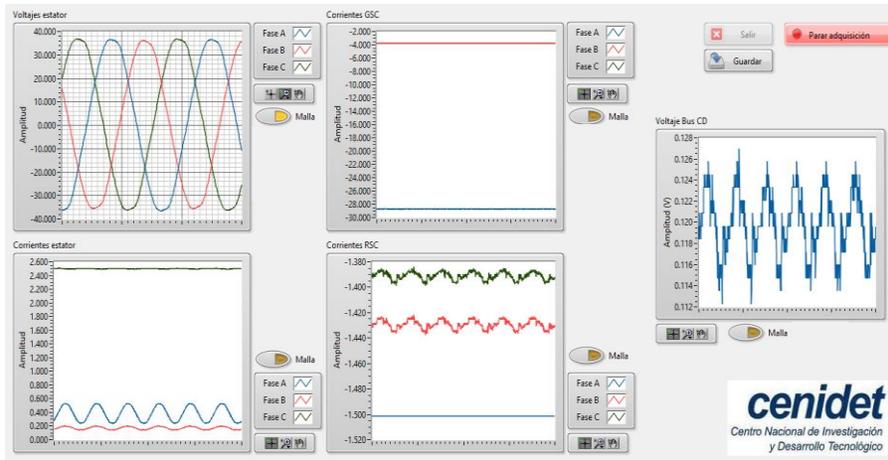


Figura 12 Captura de pantalla del sistema de adquisición.

Este equipo > TI10667800E (C:) > Datos_SAE

Nombre	Fecha de modifica...	Tipo	Tamaño
2017-06-01_11.10.49.csv	01/06/2017 11:10	Archivo de valores...	8,817 KB
2017-06-06_09.48.11.csv	06/06/2017 09:48	Archivo de valores...	52,423 KB
2017-06-07_17.01.20.csv	07/06/2017 17:02	Archivo de valores...	137,012 KB
2017-06-07_17.02.21.csv	07/06/2017 17:03	Archivo de valores...	117,712 KB

Figura 13 Archivos guardados del VI.

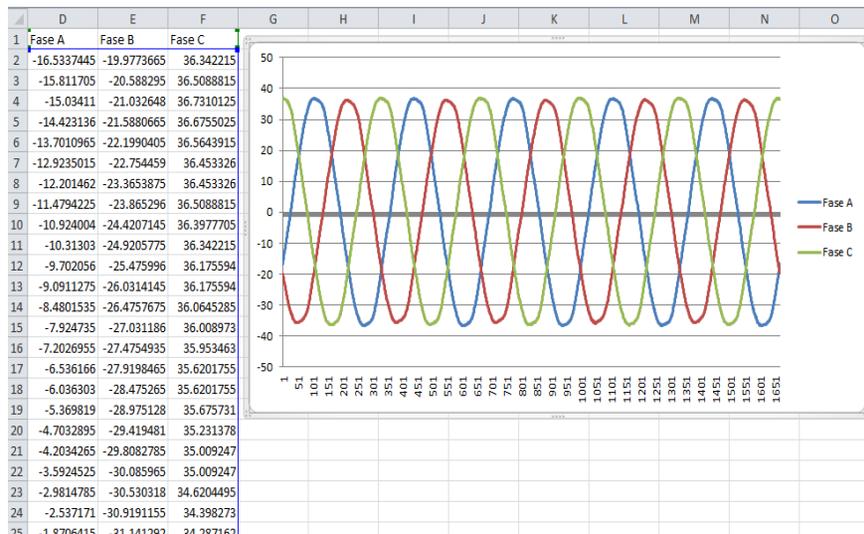


Figura 14 Graficado en Microsoft Office Excel de una de las pruebas realizadas.

4. Discusión

Los resultados expuestos mostraron el correcto funcionamiento del VI para medir y graficar el voltaje de estator del WECS, como trabajo futuro se probará la medición de todas las variables para las que fue diseñado el VI en cuanto se tenga funcionando al WECS en su totalidad.

Con respecto al diseño del instrumento virtual en la tarjeta myRIO, a continuación, se muestran unas recomendaciones:

- Para la selección de la computadora que ejecutará el programa de adquisición, se recomienda que tenga un procesador de desempeño igual o superior al *Intel Core* o *AMD A-Series*. Esto debido a que la operación de guardado en archivo ocupa muchos recursos de la *CPU* y se requiere un procesador que sea capaz de realizar este proceso y al mismo tiempo, presentar las mediciones en pantalla.
- La tarjeta *NI myRIO* permite que en lugar de la conexión USB se pueda utilizar la conexión *Wi-Fi* con su tarjeta incorporada, sin embargo, no se aconseja utilizarla en lugares donde existan numerosos puntos de acceso inalámbricos. Esto provoca que ocurra una interferencia en la comunicación inalámbrica y que eventualmente el procesamiento de los datos se retrase, la cola *FIFO* se llene y genere problemas de desbordamiento.

5. Conclusiones

El uso de la instrumentación virtual es una herramienta importante para el monitoreo de variables en un sistema, debido a que dichos instrumentos virtuales se pueden diseñar de acuerdo con las necesidades particulares de cada sistema. Estos sistemas de instrumentación virtual son también convenientes para guardar en un archivo los datos del monitoreo para que estos puedan ser revisados a detalle más adelante. En este trabajo el desarrollo de la instrumentación virtual se hizo pensado para pruebas en un prototipo de laboratorio de WECS, basado en un DFIG y un convertidor back to back. Los requerimientos de intercambio de información entre este sistema real y la computadora deben llevarse a cabo en tiempo real y sin pérdida de información; puesto que, por un lado, los análisis de

pos-procesamiento de datos dependen de la confiabilidad de estos; y por otro lado, la definición de referencias que se hagan desde la computadora hasta los controladores físicos, también debe ser confiable. En este sentido, en este artículo se mostró el diseño de un instrumento virtual implementado en el software LabVIEW y la tarjeta myRIO, a través del cual se hizo el adecuado acondicionamiento de lo que se considera las variables eléctricas más importantes del WECS, la captura y correcta visualización de estas, así como su almacenamiento en un archivo. Los resultados mostrados demuestran que la adquisición de señales se llevó a cabo de manera correcta, es decir, sin pérdida de información. Los resultados obtenidos en esta primera etapa de desarrollo son adecuados para posteriormente buscar la implementación del control del convertidor back to back de manera adicional en el hardware de la tarjeta myRIO.

6. Bibliografía y Referencias

- [1.] Aguilar, O., Tapia, R., Valderrabano, A., & Minor, H. Design and performance comparison of PI and adaptive current controllers for a WECS. *IEEE Latin America Transactions*, Vol.13, No.5, pp.1361–1368, 2015.
- [2.] Calderón, G., Mina, J., & López, A. Modelado y simulación de un Sistema de Conversión de Energía Eólica de velocidad variable interconectado a la red eléctrica . XVI Congreso Latinoamericano de Control Automático, 2014.
- [3.] Ayaz, M., Colak, I., & Bayindir, R. MATLAB/GUI Based Wind Turbine Generator Types on Smart Grid Systems. *IEEE International Conference on Renewable Energy Research and Applications (ICRERA)*, Vol. 5, pp. 1158–1162, 2016.
- [4.] Jiménez, E. A. S. Análisis del diseño y control de un generador trifásico doblemente alimentado. Universidad de Chile facultad de ciencias físicas y matemáticas departamento de ingeniería eléctrica, 2012.
- [5.] Karhe, R. R., Patil, C. S., & Patil, M. S. Real Time Data Acquisition and Home Parameters Monitoring using LabVIEW. *International Journal of Advanced Research in Computer Engineering & Technology (IJARCET)*, Vol.2, No.3, pp. 979–983, 2013.

- [6.] National Instruments. NI myRIO-1900 User Guide, 2013.
- [7.] Osorio, J. E., Perez Ramirez, J. D., & Rodriguez Barrera, M. A. Implementacion de un sistema de adquisicion de datos para monitorear una máquina de corriente directa. *Revista Tecura*, Vol.14, No. 27, 2010.
- [8.] Rajarajan, R., Mohanraj, M. R., & Prabhakaran, B. Real-Time Simulation System of Wind Power Based On Virtual Instrumentation. *International Journal of Engineering Research & Technology (IJERT)*, Vol. 2, No. 11, pp. 4169–4176, 2013
- [9.] Saa, J. F. D., Vallejo, E., & Torres, J. Diseño y Construcción de un Sistema de adquisición y Visualización de Señales Electromiográficas. *Fifth LACCEI International Latin American and Caribbean Conference for Engineering and Technology (LACCEI'2007)*, 2007.
- [10.] Sawin, J. L., Seyboth, K., & Sverrisson, F. *Renewables 2016: Global Status Report*, 2016.
- [11.] Soria Olivas, E., Martínez Sober, M., Francés Villora, J. V., & Camps Valls, G. *Tratamiento Digital de Señales: Problemas y Ejercicios Resueltos*. Universitat de Valencia, Vol. 53. Valencia, España, 2003.
- [12.] Topor, M. Wind Turbine Emulator Development Using Labview FPGA. *International Journal of Emerging Engineering Research and Technology*, Vol. 3, No. 7, pp. 13–21, 2015.