

Evaluación y mejora a tarjeta USB de adquisición de datos para LabVIEW

Evaluation and improvement to USB data acquisition card for LabVIEW

DELGADO-GUERRERO, Sergio Humberto^{†*}, LOPEZ-ALVAREZ, Yadira Fabiola, JARA-RUIZ, Ricardo y GALLEGOS-RAMIREZ, José Luis

Universidad Tecnológica del Norte de Aguascalientes

ID 1^{er} Autor: *Sergio Humberto, Delgado-Guerrero* / ORC ID: 0000-0003-2521-5887, Researcher ID Thomson: V-1747-2018, CVU CONACYT ID: 240475

ID 1^{er} Coautor: *Yadira Fabiola, Lopez-Alvarez* / ORC ID: 0000-0002-9041-1908, Researcher ID Thomson: T-1555-2018, CVU CONACYT ID: 375952

ID 2^{do} Coautor: *Ricardo, Jara-Ruiz* / ORC ID: 0000-0001-7725-4138, Researcher ID Thomson: T-1532-2018, CVU CONACYT ID: 630276

ID 3^{er} Coautor: *José Luis, Gallegos-Ramirez* / ORC ID: 0000-0002-9932-4974, CVU CONACYT ID: 240474

DOI:10.35429/JOCT.2019.9.3.25.31

Recibido 16 de Enero, 2019; Aceptado 30 Marzo, 2019

Resumen

La adquisición de datos se volverá el insumo principal para el crecimiento de la I4.0 y el IOT, por ello se presenta una evaluación y propuestas de mejora a la tarjeta de adquisición de datos USB para LabVIEW reportada en la revista de Sistemas Computacionales y TIC's en marzo de 2016. Es popular la utilización de tarjetas Arduino para la adquisición de datos, pero su arquitectura de hardware y estructuras de programación ya resueltas, impide que sus usuarios desarrollen un conocimiento propio. La tarjeta reportada previamente tiene la desventaja de estar centrada en un único SubVI que dificulta la interconexión de elementos y la generación de esquemas complejos de programación. Buscando la facilidad de uso, se presenta un rediseño del PCB, funcional en Windows 10 y que permite su conexión directa a protoboard; además se desarrollaron SubVI's individuales para las tareas de adquisición de datos, que pueden utilizarse repetidamente en LabVIEW y pueden integrarse a sus librerías. Se ha creado una página web en donde se comparten los archivos en código libre, diagramas, ejemplos y otros tutoriales. La nueva tarjeta y sus librerías son más simples de utilizar, tienen una mayor versatilidad y posibilitan la interacción con hardware periférico.

Tarjeta de adquisición de datos, SubVI, PCB

Abstract

The acquisition of data will become the main factor for the growth of the I4.0 and the IOT, so an evaluation and proposals for improvement to the USB data acquisition card for LabVIEW, reported in the journal of Computational Systems and ICT's in March 2016, are presented. The use of Arduino for the acquisition of data is popular, but its hardware architecture and programming structures already resolved, prevents its users from developing their own knowledge. The previously reported card has the disadvantage of being centered on a single SubVI that hinders the interconnection of elements and the generation of complex programming schemes. Looking for ease of use, a redesign of the PCB functional in Windows 10 and that allows its direct connection to protoboard has been done; in addition, individual SubVI's were developed for data acquisition tasks, which can be used repeatedly in LabVIEW and which can be integrated into their libraries. A web page has been created where files are shared in free code, diagrams, examples and other tutorials. The new card and its libraries are simpler to use, have greater versatility and allows interaction with peripheral hardware.

Data acquisition card, SubVI, PCB

Citación: DELGADO-GUERRERO, Sergio Humberto, LOPEZ-ALVAREZ, Yadira Fabiola, JARA-RUIZ, Ricardo y GALLEGOS-RAMIREZ, José Luis. Evaluación y mejora a tarjeta USB de adquisición de datos para LabVIEW. Revista de Tecnologías Computacionales. 2019. 3-9: 25-31

* Correspondencia del Autor (Correo electrónico: sergio.delgado@utna.edu.mx)

† Investigador contribuyendo como primer autor.

Introducción

Scalabre, O. (2017) afirma que la industria 4.0 tiene el potencial de convertirse en la próxima revolución industrial; sin embargo, en un país tercermundista, con una cultura de recepción de tecnología y no de generación de la misma, a mayor tecnología aplicada a los procesos productivos, a mayor tecnología aplicada y adoptada por sólo algunos estratos de usuarios, mayor será la brecha social y socioeconómica en dicho país; es necesario evitar los errores de las tres primeras revoluciones industriales en donde los beneficios y crecimientos fueron limitados, no a ciertos países sino que incluso dentro de los países líderes de las mismas, fue acotado a ciertas regiones de los mismos. Podremos presumir que vivimos y fuimos parte de la 4ta Revolución Industrial cuando el grado de inserción e inclusión de las tecnologías propias de la I4.0 alcance índices iguales al grado de alfabetismo, por país, región, o incluso a nivel mundial.

Es por ello que esta nueva versión de la tarjeta USB pretende impulsar el desarrollo nacional de tecnología disminuyendo la dependencia de otras tecnologías extranjeras; esto a través de un diseño de hardware que resulta familiar para el usuario de Arduino y una página web con toda la información disponible y libre para cualquier persona.

Las secciones desarrolladas en este artículo son metodología, en dónde se mencionan las bases para la selección de mejoras a implementar y resultados con varias subsecciones. En la subsección Diseño de la tarjeta USB se presenta el nuevo hardware y la descripción de los pines y su funcionalidad; en la sección Controlador para la tarjeta USB en Windows se muestra detalladamente el procedimiento de instalación del controlador para Windows 10, previamente sólo se había desarrollado para la versión de Windows 8; en la sección Actualización de las librerías y SubVI en LabVIEW se han integrado descripciones detalladas de cada una de los nuevos SubVI programados y finalmente en la sección de Ejemplos de aplicación se incluyen varias de las posibilidades y capacidades de esta nueva tarjeta y su interacción con LabVIEW.

Finalmente se puede mencionar que, a través de este trabajo, un usuario con conocimientos básicos de electrónica, programación o mecatrónica, puede fácilmente crear su propia tarjeta de adquisición de datos por USB y comenzar a utilizarla en LabVIEW, aprovechando sus vastas ventajas.

Metodología a desarrollar

Para decidir sobre el nuevo diseño y la viabilidad y ventajas qué debería tener éste, se realizó un análisis de otras tarjetas que existen actualmente y que fueron reportadas como artículos publicados; pero principalmente se revisaron sus aplicaciones en la adquisición de datos y su orientación hacia la industria 4.0 hacia IoT.

Jian-xun, W. (2011) reporta un sistema de adquisición de datos no limitado a la adquisición de datos, sino también capaz de su procesamiento; esta funcionalidad es esencial en sistemas orientados a la I4.0. Una de las ventajas de los Arduino es la cantidad de librerías, ejemplos y tutoriales que se pueden encontrar en Internet, además de la cantidad de hardware que, debido a lo anterior, se vuelve prácticamente *plug&play*. Por ello se decidió diseñar una nueva tarjeta USB, más sencilla y que continuara con una interfaz USB hacia LabVIEW ya que el equivalente de librerías, ejemplos y tutoriales que de National Instruments se puede encontrar en Internet es prácticamente el triple de lo que se puede hallar para Arduino y el hardware podría ser menor en cantidad pero no en calidad y sobre todo, está orientado y dedicado completamente a la industria de procesos y de manufactura; es por ello que el usuario de esta tarjeta USB se podrá familiarizar con código aplicable en soluciones industriales.

Hay referencias de comunicación de Arduino con LabVIEW, pero dado que utiliza protocolo RS232 a través del puerto USB su velocidad es mínima comparada con la del sistema reportado en este trabajo:

Sistema	Comunicación y velocidad
Arduino	Serial 9600 bits / s
Tarjeta USB 2.0	USB a través de PIC en configuración HID 64 kBytes / s

Tabla 1 Comparativa de velocidad de comunicación con LabVIEW entre Arduino y la tarjeta USB 2.0

Fuente: *Elaboración Propia*

Resultados

Diseño de la tarjeta USB

Se ha rediseñado la tarjeta USB para utilizarse en conjunto con un *proto-board* pero sin quitar muchos espacio a éste; el objetivo fue hacerla más simple, con un mínimo de componentes y que resulte familiar a los usuarios que están acostumbrados a las tarjetas Arduino; sus pines permiten la conexión de alambres hacia un *proto-board*. Sigue basada en un microcontrolador PIC18F2550® y utiliza un protocolo de comunicación USB 2.0 El tamaño total de la nueva tarjeta es 5.5 cm X 2.5 cm.

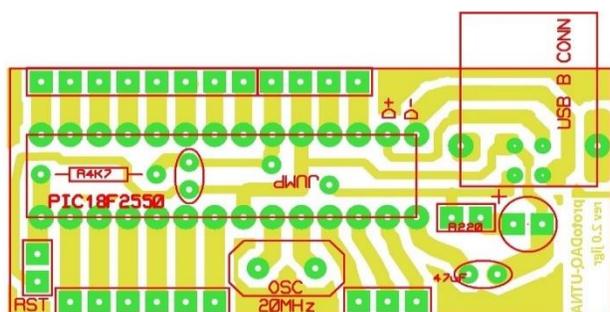


Figura 1 Diseño del PCB de la tarjeta USB 2.0 Fuente: Elaboración Propia



Figura 2 Tarjeta USB 2.0 Fuente: Elaboración Propia

Los componentes y la distribución de pines se muestran a continuación:

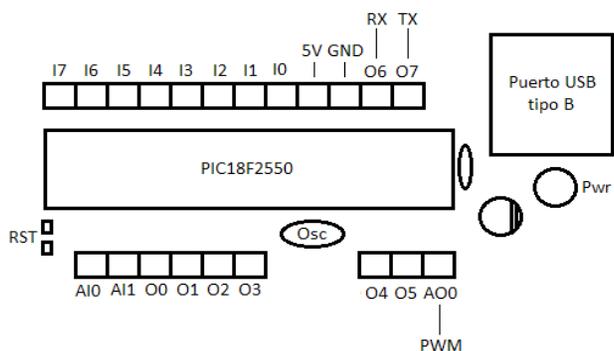


Figura 3 Ubicación de elementos y descripción de pines para la tarjeta USB 2.0 Fuente: Elaboración Propia

El listado y funcionalidad de pines es la siguiente:

Pin	Funcionalidad
AI0	Entradas analógicas 0 y 1.
AI1	Pueden recibir voltajes entre 0 y 5 V y estos se traducirán a valores digitales entre 0 y 255
O0 .. O7	Salidas digitales 0 .. 7. Sus estados pueden ser valores lógicos 1 ó 0, equivalentes a 5 y 0 V
AO0 / PMW	Salida PWM. Su salida es un tres de pulsos modulados con un ancho desde 0 hasta 100%, traducibles a través de un capacitor a voltajes desde 0 hasta 5 V.
RX	Pin compartido con O7. Es el pin de recepción de datos en protocolo serial RS232
TX	Pin compartido con O7. Es el pin de transmisión de datos en protocolo serial RS232
5V	Pin con 5V constantes
GND	Pin de referencia a GND
I0 .. I7	Entradas digitales 0 .. 7. Sus estados pueden ser valores lógicos 1 ó 0, dependiendo si se les conecta un voltaje de 5 ó 0 V

Tabla 2 Listado de pines y funcionalidad para la tarjeta USB 2.0 Fuente: Elaboración Propia

El microcontrolador PIC18F2550 debe programarse con el archivo USB 2550 ago2019 v1.hex

Controlador para la tarjeta USB en Windows 10

Se ha actualizado el controlador o driver para la tarjeta USB, para que sea funcional en Windows 7 Windows 8 y Windows 10; el archivo *daq_usb_2.0.inf* es el nuevo controlador. Estas versiones de Windows también lo detectan como un driver no firmado digitalmente por lo que los siguientes pasos se deben realizar previo a su instalación. Presione "Win + X", navegue hasta "Apagar" y luego "Shift + clic izquierdo" en la opción "Reiniciar".

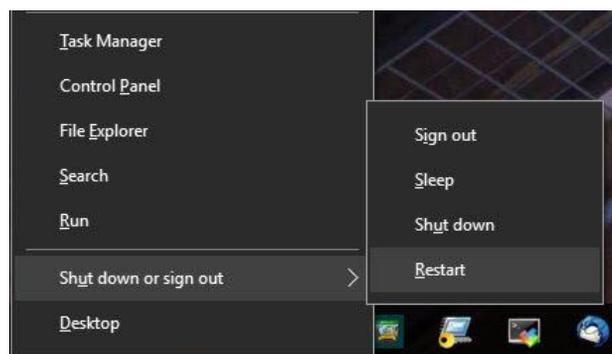


Figura 4 Pantalla para reiniciar Windows 10 Fuente: Elaboración Propia

La acción anterior reiniciará su sistema y lo llevará al menú de Arranque avanzado. Aquí, seleccione la opción "Solución de problemas".



Figura 5 Selección de Solución de problemas en el reinicio de Windows 10

Fuente: *Elaboración Propia*

En la sección Solución de problemas, seleccione la opción "Opciones avanzadas".



Figura 6 Selección de Opciones Avanzadas. Fuente: *Elaboración Propia*

Ahora, seleccione "Configuración de inicio".



Figura 7 Selección de Configuración de inicio"

Fuente: *Elaboración Propia*

La opción Configuración de inicio le permitirá iniciar su sistema Windows en diferentes modos. Simplemente haga clic en el botón "Reiniciar" para continuar.



Figura 8 En esta pantalla simplemente se hace clic en [Reiniciar]

Fuente: *Elaboración Propia*

Dado que necesitamos instalar controladores sin firmar, presione F7 en su teclado para seleccionar la séptima opción "Desactivar la aplicación de firma del controlador".

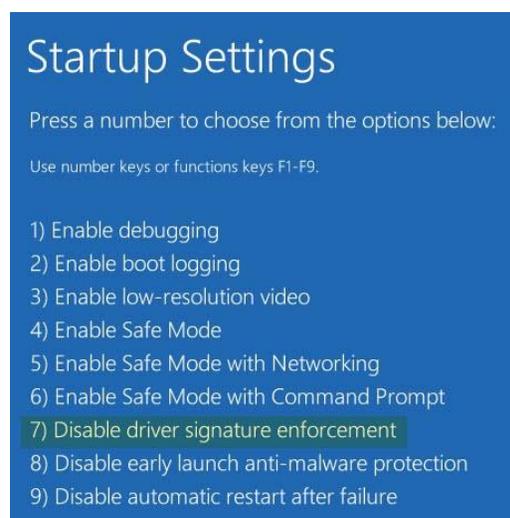


Figura 9 Opción a seleccionar para deshabilitar el requerimiento de controladores firmados

Fuente: *Elaboración Propia*

Cuando presione esta tecla, su sistema se iniciará en Windows y le permitirá instalar el controlador *daq_usb_2.0.inf* haciendo clic derecho sobre éste y eligiendo la opción Install. No debe preocuparse de este ajuste realizado, ya que después de la instalación y reinicio de su sistema, la firma del controlador se habilitará automáticamente desde el próximo reinicio.

Debe mencionarse que este no es el único método para que Windows 10 permita la instalación de controlares no firmados, pero este que se explica sí es el más sencillo.

Configuración de la tarjeta en LabVIEW

Una vez instalado el controlador se debe ejecutar el NI MAX para configurar la tarjeta USB y crear su alias de forma que pueda ser utilizada en LabVIEW; desde esta aplicación también se puede probar la funcionalidad básica de la misma. Los pasos para la configuración son los siguientes:

Abrir NI MAX e indicarle que busque el hardware recién conectado.

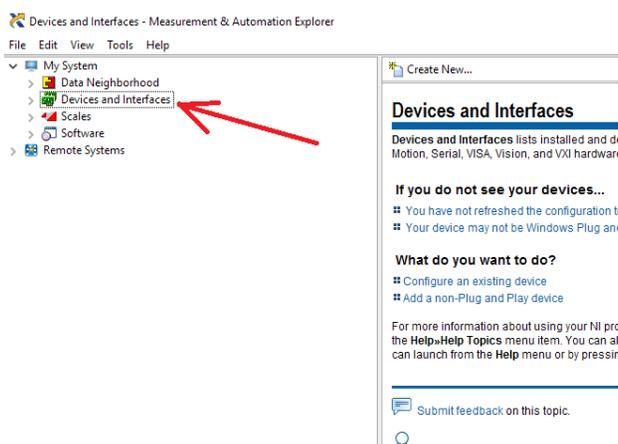


Figura 10 Selección de Dispositivos e interfaces en NI MAX
Fuente: Elaboración Propia

Se asigna un nombre o alias a la tarjeta USB 2.0 Este nombre será utilizado en todos los programas en LabVIEW; es recomendable utilizar un nombre corto y simple. Enseguida debemos abrir el VISA Test Panel, que nos permitirá verificar la funcionalidad básica de la tarjeta; esto es probar que esté recibiendo y enviando datos.

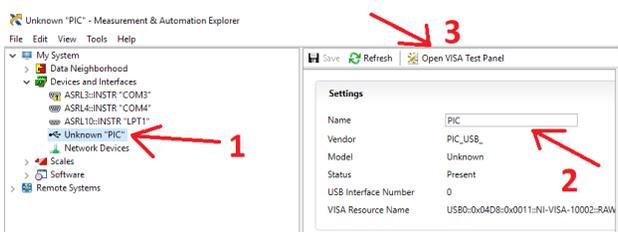


Figura 11 Introducción de nombre para la tarjeta USB 2.0 En la figura se asignó el nombre “PIC” Fuente: Elaboración Propia

En la ventana que se abre, dirigirse al apartado Input/Output (1); enseguida se debe anotar que se leerán 8 bytes (2); a continuación se ejecutan las acciones de escritura (3) y lectura (4) en ese orden. La salida que se debe obtener (5) cuando no se tiene nada conectado a la tarjeta USB 2.0 se muestra también en la figura 12.

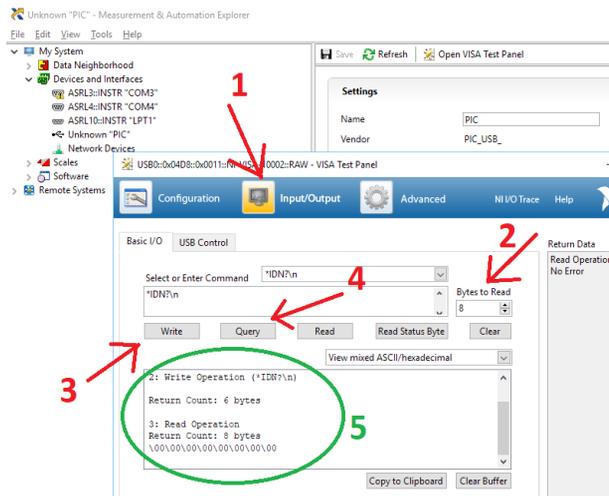


Figura 12 Ejecución de las tareas de Escritura y Lectura con 8 bytes
Fuente: Elaboración Propia

Una vez concluidos estos pasos la tarjeta está configurada para usarse en LabVIEW. Es importante señalar que estos pasos no deben repetirse, aunque la tarjeta se desconecte, tampoco si LabVIEW se cierra ni tampoco si se apaga la computadora; la única circunstancia bajo la cual este procedimiento deberá repetirse es si la tarjeta se conecta a un puerto USB diferente a donde se había hecho la configuración.

Para desconectar la tarjeta USB 2.0 de la computadora es recomendable removerla de forma segura desde el menú de la barra de tareas; la tarjeta siempre se identificará como DAQ_USB_2.0

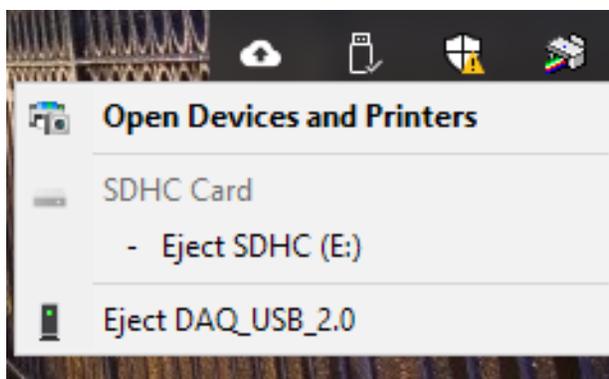


Figura 13 Desconexión de la tarjeta USB 2.0 de forma segura
Fuente: Elaboración Propia

Actualización de las librerías y SubVI's en LabVIEW

En la versión anterior de este desarrollo se tenía un único SubVI que centralizaba todas las funciones de la tarjeta USB y que además no se podía repetir; esto dificultada su utilización en LabVIEW. Para la tarjeta USB 2.0 se han programado una serie de SubVI específicos para cada una de las funciones principales de la tarjeta, con nombres e íconos que facilitan mucho su uso, además pueden repetirse tantas veces sea necesario dentro de un programa en LabVIEW.

SubVI	Funcionalidad
 Escribe_digital.vi	LabVIEW escribe un dato en el puerto digital de salida de la tarjeta.
 Escribe_serial.vi	LabVIEW escribe un dato que se transmite por RS232 a 9600 baudios por el pin TX.
 Escribe_PWM.vi	LabVIEW escribe un dato que se traduce a una salida de nivel PWM en el pin PWM.
 Lee_entrada_A...	LabVIEW lee el valor digital de la entrada analógica A0 de la tarjeta.
 Lee_entrada_A...	LabVIEW lee el valor digital de la entrada analógica A1 de la tarjeta.
 Lee_entrada_di...	LabVIEW lee el valor digital del puerto digital de entrada de la tarjeta.

Tabla 3 Descripción de los SubVI creados para el manejo para la tarjeta USB 2.0
Fuente: *Elaboración Propia*

Para facilitar aún más el uso de estas librerías se recomienda al usuario copiarlas a la carpeta donde se instaló LabVIEW: National Instruments/LabVIEW 2018/user.lib de esta manera se integrarán al menú de Funciones en la ventana de diagrama de bloques en LabVIEW.

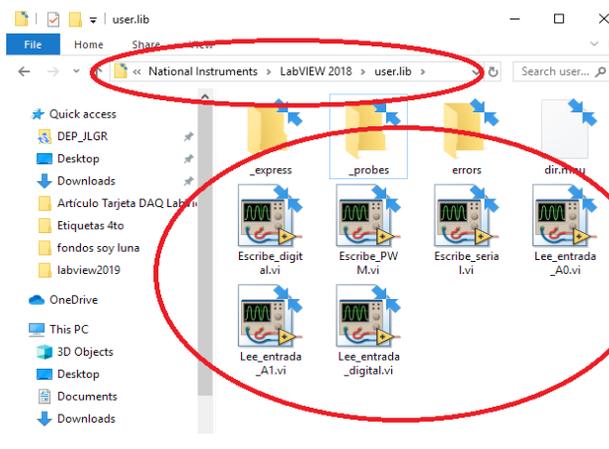


Figura 14 Ubicación de los SubVI en el explorador de Windows
Fuente: *Elaboración Propia*

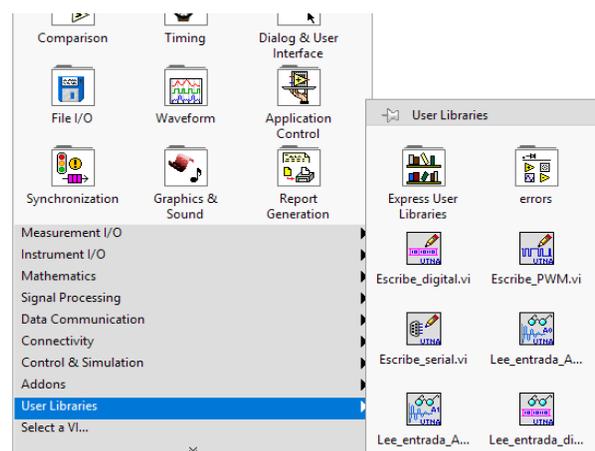


Figura 15 Integración de los SubVI en las librerías de funciones en LabVIEW
Fuente: *Elaboración Propia*

Ejemplos de ejercicios

Son muchos los ejercicios y ejemplos que se han realizado y probado con la nueva tarjeta USB 2.0 y sus nuevos SubVI, todos ellos se irán subiendo a la página web. El siguiente caso es un mero ejemplo en donde se muestran todos los SubVI interactuando, tanto los de lectura como los de escritura y además 3 instancias de una lectura analógica.

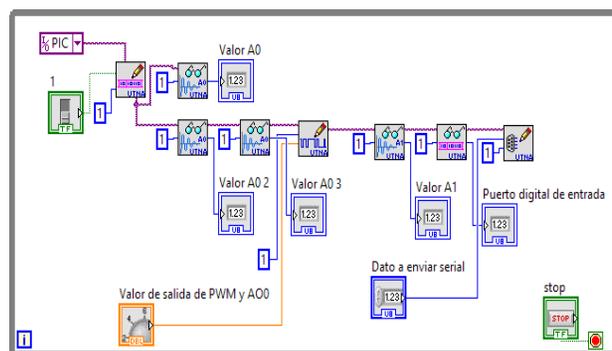


Figura 16 Utilización de los SubVI en un ejercicio
Fuente: *Elaboración Propia*

Distribución libre de información

A través del servicio Wix® se ha creado una página web gratuita y de libre acceso en donde se han colocado todos los diagramas, programas, librerías y ejemplos hasta ahora creados para esta tarjeta USB 2.0. El siguiente paso es seguirla alimentado con nuevos ejercicios y mayores utilerías como un foro en donde se comparta información y casos de éxito de parte de los usuarios.

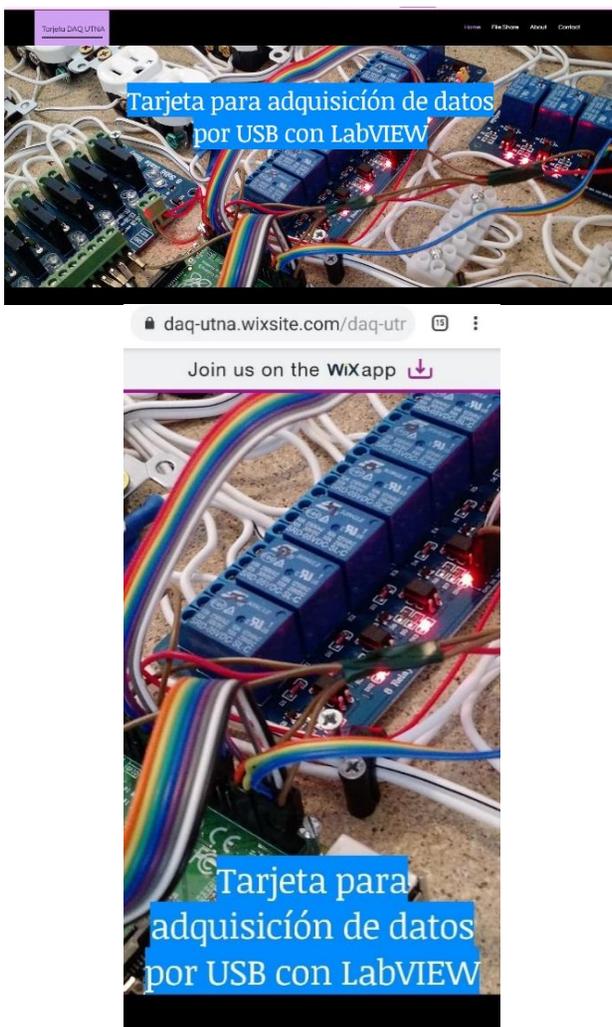


Figura 17 Página web donde se comparte la información de la tarjeta USB 2.0, vista en PC y en smarthphone
Fuente: *Elaboración Propia*

Agradecimiento

Los autores agradecen el apoyo de la Universidad Tecnológica del Norte de Aguascalientes por el apoyo otorgado durante el desarrollo de este artículo.

Conclusiones

Se han propuesto e implementado mejoras a la tarjeta de adquisición de datos reportada en 2016, en base a evaluaciones de aquella y a tendencias de aplicación buscadas como la Industria I4.0.

Estas mejoras incluyen: actualización de hardware a una versión mínima de elementos y compatible directamente con un *protoboard*, actualización de controlador para Windows 10, creación de SubVI específicos para cada tarea de la tarjeta.

Propiciando una arquitectura abierta y bajo la filosofía de software libre, todos los archivos para la construcción y puesta en marcha de la tarjeta USB 2.0, así como ejemplos de uso, se han subido a una página web libre.

Las pruebas realizadas a la nueva tarjeta con sus nuevas SubVI muestran una configuración razonable del hardware y facilidad de operación durante la programación en LabVIEW.

La adquisición de datos desde la tarjeta USB 2.0 y hacia LabVIEW, se desempeña a una alta velocidad.

Referencias

SCALABRE, O. (2017). Industry 4.0: The Next Industrial Revolution. Boston Consulting Group (BCG)[online][vid. 2017-12-08]. www.bcg.com/capabilities/operations/embracing-industry-4.0-rediscovering-growth.aspx.

Jian-xun, W. (2011). Design of Data Acquisition System Based on USB Interface and LabVIEW.