

**DISEÑO Y CONSTRUCCIÓN DE UNA TARJETA MULTICANAL PARA LA
ADQUISICIÓN DE SEÑALES FISIOLÓGICAS**

WILMER ARLEY VIANCHA CORREGIDOR

**UNIVERSIDAD INDUSTRIAL DE SANTANDER
FACULTAD DE INGENIERÍAS FÍSICO-MECÁNICAS
ESCUELA DE INGENIERÍAS ELÉCTRICA, ELECTRÓNICA Y DE
TELECOMUNICACIONES
BUCARAMANGA
2015**

**DISEÑO Y CONSTRUCCIÓN DE UNA TARJETA MULTICANAL PARA LA
ADQUISICIÓN DE SEÑALES FISIOLÓGICAS**

WILMER ARLEY VIANCHA CORREGIDOR

**TRABAJO DE GRADO PRESENTADO COMO REQUISITO PARCIAL PARA
OPTAR AL TÍTULO DE INGENIERO ELECTRÓNICO**

DIRECTOR

**JAIME GUILLERMO BARRERO PÉREZ,
INGENIERO ELÉCTRICO, M.I**

CODIRECTOR

**SALVADOR PACHECO
INGENIERO ELECTRÓNICO**

**UNIVERSIDAD INDUSTRIAL DE SANTANDER
FACULTAD DE INGENIERÍAS FÍSICO-MECÁNICAS
ESCUELA DE INGENIERÍAS ELÉCTRICA, ELECTRÓNICA Y DE
TELECOMUNICACIONES
BUCARAMANGA**

2015

AGRADECIMIENTOS

Reconocimiento especial al Profesor Jaime Barrero quien estuvo como director de este proyecto y gracias a su aporte, colaboración y conocimientos hicieron posible la realización de este.

Agradezco todo el apoyo, colaboración, ayuda y asesoría al profesor Salvador Pacheco quien figura como codirector del proyecto y actualmente es estudiante de maestría de la Universidad Industrial de Santander, pues siempre estuvo facilitando el trabajo, colaborando con las pruebas así como también dando aportes y motivación para la realización del mismo.

CONTENIDO

	Pág.
INTRODUCCIÓN	10
1. DEFINICIÓN DEL PROBLEMA	11
1.1. OBJETIVOS	12
1.2. OBJETIVO GENERAL.....	12
1.3. OBJETIVOS ESPECÍFICOS	12
1.4. JUSTIFICATION.....	13
1.5. ALCANCE	14
2. DISEÑO Y CONSTRUCCIÓN TARJETA DE ADQUISICIÓN DE DATOS	15
2.1. MICROCONTROLADOR.....	16
2.2. MÓDULO USB	20
2.3. CASO USB CLASE CDC	21
2.4. CASO USB CLASE BULK.....	22
3. FIRMWARE PARA EL MICROCONTROLADOR	24
3.1. CASO USB CLASE CDC	24
3.2. CASO USB CLASE BULK.....	26
4. DISEÑO E IMPLEMENTACIÓN INTERFAZ GRÁFICA	27
4.1. CASO USB CLASE CDC	27
5. PRUEBAS Y FUNCIONAMIENTO DEL SISTEMA DE ADQUISICIÓN DE DATOS.	32
6. CONCLUSIONES Y RECOMENDACIONES	38
BIBLIOGRAFIA	40

LISTADO DE FIGURAS

	Pág.
Figura 1: Diagrama de bloques tarjeta de adquisición de datos	15
Figura 2: Núcleo de la familia ColdFire V1	16
Figura 3: Interfaz Panel de control envío de datos.....	24
Figura 4: Diagrama de flujo tarjeta adquisición de datos.	25
Figura 5: Interfaz de usuario Adquisición de datos.	27
Figura 6: Interfaz gráfica Lectura de datos (Tipo de archivos .tdms)	28
Figura 7: Mensaje: Ventana Grafica: ¿Desea Guardar imagen?	30
Figura 8: Visualización de gráficas.	30
Figura 9: Recepción de señales en tarjeta de adquisición de datos, señales provenientes de sensores en placa de desarrollo.....	34
Figura 10: Visualización de señales tomadas por Generador de señales.	35
Figura 11: Visualización de señales provenientes de sensores (Acelerómetro de tres ejes (X, Y, Z), potenciómetro).	35

RESUMEN

Título: DISEÑO Y CONSTRUCCIÓN DE UNA TARJETA MULTICANAL PARA LA ADQUISICIÓN DE SEÑALES FISIOLÓGICAS¹

Autor: WILMER ARLEY VIANCHA CORREGIDOR²

Palabras Clave: Adquisición de datos, puerto USB, señales fisiológicas, microcontrolador, interfaz gráfica.

Descripción:

Este trabajo de grado aborda el proceso de diseño y construcción de una tarjeta de adquisición de datos, que permitirá la captura de señales fisiológicas de baja frecuencia de manera simultánea, lo anterior con el fin de presentar un equipo que permita realizar monitoreo y análisis a diferentes señales fisiológicas en un paciente al mismo tiempo, con esto se tomaran y visualizaran datos en tiempo real sobre la respuesta al organismo de una persona frente a un determinado estímulo a estudio médico. Para lograrlo, esta tarjeta contará con hasta diez canales de adquisición de señales fisiológicas que serán tomadas secuencialmente de los sensores previamente ubicados en el área del paciente a examinar, serán digitalizadas y adecuadas en la tarjeta de adquisición de datos para luego ser enviadas a través del puerto USB al computador, en el cual se podrán visualizar por medio de una interfaz diseñada en Labview y dedicada para tal fin.

Con el desarrollo e implementación del conjunto tarjeta de adquisición de señales fisiológicas + interfaz para visualización y almacenamiento de datos, se lograra realizar un estudio completo al comportamiento y estado de los diferentes órganos, correlación y respuesta ante un estímulo en un paciente, con lo anterior se ayudara a fortalecer las prácticas y desarrollo de actividades a los estudiantes de medicina de la Universidad Industrial de Santander.

¹ Trabajo de grado

² Facultad de ingenierías Físico-Mecánicas. Escuela de ingenierías Eléctrica, Electrónica y de Telecomunicaciones.

Director: Jaime Guillermo Barrero Pérez, Ingeniero Eléctrico, M.I. Codirector: Salvador Pacheco, Ingeniero Electrónico.

ABSTRACT

Title: DESIGN AND CONSTRUCTION OF A MULTICHANNEL CARD FOR THE ACQUISITION OF PHYSIOLOGICAL SIGNS³

Author: WILMER ARLEY VIANCHA CORREGIDOR⁴

Keywords: Data Acquisition, USB port, physiological signs, microcontroller, graphical interface.

Description:

This work of end of degree deal with the process of designing and construction of a card of data acquisition, which will allow the capture of simultaneous low frequency physiological signs, the previous to present an equipment that allows monitoring and analyzing different physiological signs at the same time in patients, using this will be taking and visualizing real time data of response on the organism of a person faced to certain stimulus or medical tests. To achieve it, the card will have ten acquisition channels of physiological signs, which will be taken sequentially from the sensors previously located in the patient to examine, that data will be digitized and adapted in the card of data acquisition to send it through the USB port to the computer, where it will be visualized within a designed interface in Labview and design for such purpose.

Using the development and implementation of the set, acquisition card of physiological signs and interface to visualize and storage of data, will be able to carry out a complete test of the performance and state of the different organs, correlation and response to a stimulus in patient, this will help to strengthen the practices and activities development of students of medicine at Universidad Industrial de Santander.

³ Bachelor Thesis

⁴ Physico-Mechanical Engineering Faculty. School of Electrical Engineering. Supervisor: Jaime Guillermo Barrero Pérez, Electrical Engineer, M.I. Co- Supervisor: Salvador Pacheco, Electronic Engineer.

INTRODUCCIÓN

Debido al incremento de enfermedades y/o factores de riesgo que ponen en peligro la calidad de vida o la salud de las personas, y ajustándose a la política de control y a la detección temprana de enfermedades, surge la necesidad de contar con un equipo que realice mediciones en tiempo real sobre las diferentes señales fisiológicas de baja frecuencia presentes en el cuerpo humano.

Este equipo, a su vez, pretende ayudar a fortalecer los laboratorios de la facultad de salud de la Universidad Industrial de Santander (UIS), pues existe la necesidad de dotarlos con dispositivos que permitan la medición de diferentes señales fisiológicas simultáneamente, y con esto realizar análisis y estudios electrofisiológicos (EEF) más completos sobre el estado de salud del paciente.

Para cumplir lo anterior, es necesario realizar una serie de procedimientos, que más adelante se expondrán con mayor detalle y que consisten, a grandes rasgos, en lo siguiente: Implementar el hardware necesario para el muestreo de señales fisiológicas a una frecuencia de muestreo de 500 Hz y de esta forma obtener el valor digital que represente a las mismas; habilitar 10 canales de adquisición de datos para la toma de señales fisiológicas en tiempo real; realizar *firmware* para el microcontrolador que permita obtener los datos de las señales fisiológicas y enviarlos a través del módulo *USB* incluido en el microcontrolador; diseñar la interfaz en Labview que permita mostrar los datos adquiridos desde el dispositivo por medio del puerto *USB*.

1. DEFINICIÓN DEL PROBLEMA

Debido al incremento de enfermedades y/o factores de riesgo que ponen en peligro la calidad de vida o la salud de las personas, y ajustándose a la política de control y la detección temprana de enfermedades, surge la necesidad de contar con un equipo que realice mediciones en tiempo real sobre las diferentes señales fisiológicas presentes en el cuerpo humano.

A su vez, este equipo pretende ayudar a fortalecer los laboratorios de la facultad de salud de la Universidad Industria de Santander (UIS), pues existe la necesidad de dotar los laboratorios con dispositivos que permitan la medición de diferentes señales electrofisiológicas simultáneamente, y con esto realizar análisis y estudios electrofisiológicos (EEF) más completos sobre la calidad de vida del paciente.

En el momento de diseño y posterior implementación de esta tarjeta de adquisición de datos, se tendrá en cuenta el concepto de portabilidad, eficiencia y bajo consumo de potencia, así como también se diseñará una interfaz amigable con el usuario.

1.1. OBJETIVOS

1.2. OBJETIVO GENERAL

- Diseñar e implementar una tarjeta de adquisición de señales fisiológicas con conexión al computador por medio del puerto USB.

1.3. OBJETIVOS ESPECÍFICOS

- Implementar el hardware necesario para el muestreo de señales fisiológicas a una frecuencia de muestreo de 500 Hz y de esta forma obtener el valor digital que represente a las mismas.
- Habilitar 10 canales de adquisición de datos para toma de señales fisiológicas en tiempo real.
- Realizar firmware para el microcontrolador que permita obtener los datos de las señales fisiológicas y enviarlos a través del módulo USB incluido en el microcontrolador.
- Diseñar la interfaz en LabView que permita mostrar los datos adquiridos desde el dispositivo por medio del puerto USB.

1.4. JUSTIFICATION

El presente trabajo se presenta como un requisito esencial exigido por la universidad para otorgar el título de pregrado, en este caso el de Ingeniero Electrónico. Lo anterior se logrará en la medida en que se apliquen las temáticas vistas a lo largo de la carrera, y se cumplan los demás requisitos solicitados por la escuela. Por otro lado y considerando lo anterior, el proyecto pretende ayudar al mejoramiento de la calidad de vida y bienestar de las personas, pues permite llevar un monitoreo en tiempo real sobre los signos vitales del paciente y con ello detectarle enfermedades o posibles problemas de salud. El mencionado proyecto se realizara por medio de la implementación de dispositivos electrónicos de alto rendimiento y bajo consumo de potencia, abordando temas como el procesamiento de señales y datos, instrumentación electrónica, monitoreo de señales por medio de software (Labview), entre otros.

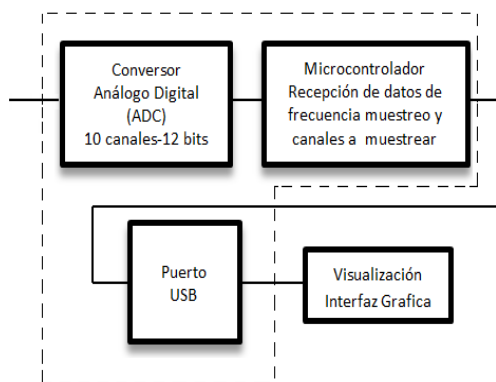
1.5. ALCANCE

El presente proyecto implementará una tarjeta de adquisición de datos, esto se logrará al recibir datos procedentes de señales fisiológicas debidamente adecuadas a niveles permitidos por el microcontrolador con los cuales se llevará un seguimiento y análisis sobre los signos vitales de una persona. Seguido a esto, se diseñará una interfaz en Labview en la cual se recibirán los datos de la tarjeta por medio del puerto USB para proceder a la visualización de los mismos. Dicha tarjeta se caracterizará por su eficiencia y portabilidad.

2. DISEÑO Y CONSTRUCCIÓN TARJETA DE ADQUISICIÓN DE DATOS

En el momento del diseño y posterior implementación de esta tarjeta de adquisición de datos, se tuvo en cuenta los conceptos de portabilidad, eficiencia, así como también se diseñó una interfaz de fácil operación que cuente con los elementos necesarios para la visualización de los mismos; ésta cuenta con un *set* de opciones que permitieran al usuario el almacenamiento de datos en dos formatos distintos (*.bmp* *.tmds*) para su posterior visualización y/o lectura y tratamiento de los datos almacenados.

Figura 1: Diagrama de bloques tarjeta de adquisición de datos

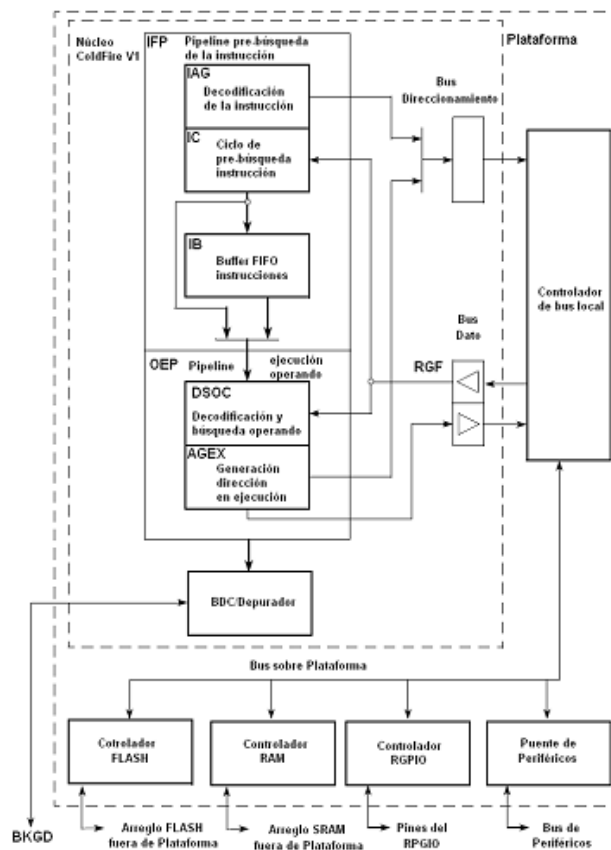


El sistema de adquisición y transmisión de datos está formado por una etapa de muestreo y digitalización (conversión análogo-digital) de datos, esta etapa podrá ser controlada y configurada desde la interfaz gráfica. Seguido a esta se encuentra la etapa de recepción de datos, al computador, la cual será la encargada de gestionar el envío de las señales digitalizadas, así como también de recibir datos por parte del usuario respecto a valores de frecuencia de muestreo y canales del *ADC* a desactivar o activar, [Figura 1]. A continuación se explica cada uno de los módulos que la conforman así como también la configuración de cada uno de ellos.

2.1. MICROCONTROLADOR

Después de realizar un sondeo al mercado de Microcontroladores correspondiente se optó por trabajar con el microcontrolador MCF51JM128 de la familia *ColdFire V1* del fabricante *Freescale* [Fig. 2], pues posee una buena relación costo/desempeño, módulo de comunicación usb integrado, así como buena resolución de conversión análogo/digital, adicional dentro de las características generales de la familia *ColdFire V1* se encuentra 3 principios importantes:

Figura 2: Núcleo de la familia ColdFire V1



- Núcleo: De tamaño reducido, la más baja disipación de potencia y el más amplio desempeño.

- Periféricos y distribución de pines: Compatibilidad total con las máquinas de 8 bits de la familia HCS08.
- Segmentación (Pipeline): Derivada de la familia *ColdFire® V2* en forma simplificada. Esta característica le proporciona al núcleo un alto desempeño.

El consumo de potencia, fue otro elemento que se tuvo en cuenta en el momento de la selección, pues contar con un dispositivo de bajo consumo ayudara a futuro a construir una versión portátil del equipo la cual podrá ser llegada a usar fuera del laboratorio.

Este microcontrolador cuenta con 2 modos primarios de operación “*run*” y “*stop*”.

En el modo de operación “*run*” el reloj del CPU del microcontrolador puede llegar a usarse al máximo (**55.33 [MHz]**), todos los periféricos están en condición de ser operados y todas sus fuentes internas están totalmente reguladas (Modo de operación *run* normal). Además, existe la posibilidad de llevar el regulador interno a estado *standby* y de esta manera ubicar la CPU en modo de bajo consumo (es importante saber que el sistema queda alimentado de manera no regulada y que todos los periféricos no usados son privados de la señal de reloj. Esto se conoce como modo de operación *modo run* en bajo consumo).

En el modo “*stop*” se encuentran 3 modos de operación (Stop2, Stop3, Stop4). En estos modos de operación el microcontrolador funciona parcialmente; algunos periféricos se encuentran apagados o se encuentran en estado de suspensión, esto con el fin de optimizar el uso de los recursos energéticos (para más detalles de estos modos de operación se recomienda al lector revisar el manual de referencia del microcontrolador MCF51jm128), estos modos de operación podrán ser configurados de acuerdo a las exigencias propias de cada proyecto o de acuerdo al criterio de cada diseñador.

Una ventaja adicional a la hora de trabajar con estos microcontroladores, es la posibilidad de realizar una tarjeta programadora de bajo costo o de fácil acceso en el mercado local que permita programar el *firmware* para la tarjeta de adquisición de datos.

A continuación se enuncian las cualidades más destacadas de este microcontrolador.

- Hasta 51 pines I/O de propósito general.
- 4 opciones de fuentes de reloj.
- Velocidad máxima de reloj de hasta 55.3 [MHz] -- desde 2.7 [V] hasta 5.5 [V]

(En la tarjeta se trabajó con una fuente de 5 [V], esta es suministrada por el modulo USB del computador)

- Puerto USB-OTG (USB *On The Go*) (Permite utilizar el modulo USB del microcontrolador como dispositivo y como Maestro.)
- Hasta 128KB de memoria FLASH de dos ciclos de acceso con bajo consumo de energía. El controlador de la FLASH permite acceso por especulación como técnica de reducción de tiempos y eficiencia en la ejecución del programa del usuario.
- Hasta 16KB de memoria RAM con un ciclo de acceso, implementada en la plataforma del procesador sobre un bus de alta velocidad.

Tanto las máquinas como los microcontroladores, miden y convierten las variables analógicas presentes en el mundo con las cuales el hombre está en continuo contacto; de allí surge la necesidad de realizar un procesamiento de

dichas variables con el fin de controlarlas. Cabe resaltar que durante el proceso de convertir señales analógicas a digitales se introducirá una inevitable pérdida de información; ésta pérdida es inherente al proceso de digitalizar las señales análogas y continuas que finalmente serán llevadas a cantidades binarias.

El módulo ADC (Convertidor Análogo-Digital) es el encargado de la toma de las señales procedentes de los sensores. El módulo ADC del microcontrolador MCF51JM128 posee 12 canales para la captura de señales y estos cuentan con una resolución máxima de 12 bits. En el momento de la configuración e inicialización de este módulo se tomó como referencia máxima de entrada el valor de 5 [V] y una referencia o voltaje mínimo de 0 [V], esto con el fin de optimizar y aprovechar el rango de excursión de la señal de entrada y que esta coincida con la ventana de conversión. Así pues, se podrán tener $2^{12} = 4096$ valores discretos posibles, e idealmente, tiene un paso mínimo de 1.22 [mV].

A la hora de realizar el muestreo de las señales adquiridas se debe tener en cuenta el teorema de Nyquist, el cual indica que cualquier señal que se quiera muestrear y en el cual no se quiera tener pérdida de información, se debe muestrear con una frecuencia mayor o igual al doble de la frecuencia máxima de la señal en cuestión.

Para el caso de las señales de baja frecuencia tratadas en el presente proyecto, se presentan dos clases de comunicación USB con las cuales se logra hacer el respectivo muestreo de las mismas. Con la clase USB-CDC⁵ (emular un puerto serial) se logra tener una frecuencia máxima de muestreo de 250 [Hz], debido a esto se genera una limitante en cuanto al tipo de señales que se pueden trabajar. En la otra clase de comunicación, USB-BULK⁶, se logra llevar la tarjeta a una frecuencia de muestreo máxima de 1 [KHz] por canal de entrada; por tanto, con este tipo de comunicación se podrán adquirir señales que no se

⁵ Communications device class

⁶ transmisiones masivas de datos

alcanzan a muestrear con la clase USB-*CDC* y que a su vez no superen un valor máximo de frecuencia de 500 [Hz].

2.2. MÓDULO USB

Este módulo es el encargado de gestionar el envío de las señales digitalizadas. Se comunica al computador mediante el protocolo de comunicación USB 2.0, que permite enviar datos con una velocidad máxima de 12 [Mb/s]. Esta velocidad es la máxima alcanzada por el microcontrolador usado.

Al momento de realizar las pruebas correspondientes al correcto funcionamiento de la tarjeta, se encontraron algunos inconvenientes respecto a la velocidad de transmisión de la misma debido a limitantes propias del sistema de cómputo y/o el software LabView con el cual se visualizan, lo anterior debido al volumen de datos a enviar; por tanto, este documento mostrara el proceso realizado a lo largo de las dos (2) clases de comunicación USB empleadas: (USB clase *CDC*, USB clase *BULK*) y con esto se entenderá la limitante de cada una de ellas.

Lo anterior debido a que el proceso se inició con comunicación USB clase *CDC* y a medida que se avanzaba en el mismo se empezó a presentar inconvenientes propios de este protocolo de comunicación. Por tanto se optó por agregar comunicación USB clase *BULK a la transferencia de datos* ya que esta permite obtener la máxima velocidad para el protocolo USB.

Pero debido al proceso iniciado con USB clase *CDC*, se decidió continuar con las dos variantes de comunicación con el fin de registrar y obtener más información sobre la comunicación USB y finalizar con una conclusión general y que esta pueda servir de soporte en la elección de envío de datos por medio de protocolo USB de acuerdo a las necesidades del proyecto que se está tratando.

A lo largo del texto se trabajarán de forma paralela los dos tipos de comunicación, con el fin de que el lector pueda apreciar las ventajas y desventajas de cada una de ellas.

2.3. CASO USB CLASE CDC

El sistema operativo Windows contiene varios *drivers* que permiten la comunicación con diferentes tipos de *hardware*. Cada uno de estos *drivers* maneja características diferentes de configuración y de comunicación. Por ejemplo, en el puerto serie se emplean dos drives en específico los cuales son:

El driver “serial.sys” el cual: controla la comunicación con los dispositivos seriales; el driver “serenum.sys”: es un filtro de más alto nivel que se encarga de enumerar y de recuperar la información de identificación para los dispositivos seriales con funcionalidad de Plug-and-Play.

En este tipo de comunicación, emulando un puerto serial, se necesita de un controlador de dispositivo (*driver*) proporcionado por Windows, pues a pesar de tener una comunicación serial, se puede llevar la velocidad de transmisión de datos a una razón de 1 [Mb/s]. Un dispositivo con tipo de conexión USB, y que se encuentra trabajando con la clase *CDC* para que pueda emular un puerto serial, requiere de un driver adicional; Este es un driver del bus que administra la comunicación entre el driver del puerto serie y el driver USB este driver es más conocido como “usbser.sys”.

El archivo “.inf” que se encuentra junto con el driver (usbser.sys), es un descriptor que contiene información necesaria para identificar los drivers de un dispositivo en particular.

La presente tarjeta puede funcionar utilizando la clase *CDC*. Dentro de las características de este tipo de comunicación se permite emular un puerto *RS232* por medio de un puerto USB.

Una de las ventajas de esta clase de comunicación USB, a la hora de trabajar con *labview*, es la no necesidad de llamar algún archivo o librería extra para poder obtener los datos de la tarjeta de adquisición de datos, lo anterior debido a que *Labview* posee bloques dedicados para la lectura/escritura con protocolo serial.

Además bajo un sistema operativo libre, en este caso Linux en su distribución *Ubuntu*, automáticamente el Sistema Operativo reconoce el dispositivo y le asigna el driver correcto. Lo podemos ver si abrimos la carpeta **/dev.**, este es el directorio donde se guardan los controladores de dispositivos en Linux. En dicho sistema operativo, a diferencia de Windows, todo se gestiona como si se tratara de un archivo. Después de conectar el dispositivo se podrá observar que se ha creado un archivo (puerto *COM* en Windows) con el nombre **ttyACM0**, ese será el puerto al que se tendrá que conectar desde la aplicación de escritorio o interfaz gráfica según sea el caso.

2.4. CASO USB CLASE BULK

Como en este caso el código programado en el microcontrolador no se ajusta a las especificaciones de alguna de las clases definidas por la gran mayoría de fabricantes de dispositivos de comunicación USB; no se podrá beneficiar del uso de los controladores de dispositivo (*driver*) que proporciona por defecto Windows, y que ya se encuentran instalados en el sistema operativo. Por lo anterior, se puede pensar en la idea de crear un driver que atienda las necesidades del firmware programado en el microcontrolador, pero debido a la complejidad de esta tarea se opta por trabajar con uno personalizado que puede ser conseguido por medio de los fabricantes de microcontroladores.

Para este proyecto se optó por trabajar con el controlador de dispositivo (*driver*) *mchpusb.sys* el cual, según las características mostradas, se ajusta a las necesidades requeridas para el diseño de la tarjeta de adquisición de datos.

En cuanto a la visualización de los datos en LabVIEW, se debe incorporar un medio que permita la comunicación entre la aplicación de usuario y la tarjeta; por tanto se debe utilizar el archivo o librería de enlace dinámico *mpusbapi.dll*. Esta actúa como una capa de software que, en este caso, por medio de funciones, facilita la comunicación entre la interfaz de usuario y el driver del dispositivo. Esta librería de enlace dinámico (.dll) se encuentra compilada en Borland C++. Para el presente caso, basta con llamar las funciones públicas de esta librería, desde nuestra interfaz de usuario, para obtener la comunicación con la tarjeta de adquisición de datos.

En este orden de ideas, aparentemente, sería más práctico la implementación del tipo de comunicación USB clase *CDC*. Esto genera, consecuentemente, la inquietud ¿Qué ventajas podría prestar la comunicación USB clase *BULK*?, pues si no representa ninguna por qué hacerlo o por qué no implementar solamente la comunicación USB clase *CDC*.

Básicamente, la ventaja de tener un driver personalizado, en vez de los genéricos que ofrece Windows, se reflejaría en una comunicación más versátil y en un número de prestaciones mayores, dentro de las cuales cabe destacar: mayor velocidad de transmisión de datos y un soporte frente a los tipos de transferencia presentes en el protocolo USB (*Control, Isócronas, Interrup y Bulk*), cosa que por el contrario no ocurrirá si se utilizan los drivers genéricos ofrecidos por Windows.

3. FIRMWARE PARA EL MICROCONTROLADOR

3.1. CASO USB CLASE CDC

El *Firmware* del microcontrolador se realizó permitiendo que este tenga comunicación en ambos sentidos. El flujo de datos del microcontrolador hacia la interfaz muestra los canales del *ADC* que el usuario previamente ha seleccionado, por defecto, el programa del microcontrolador envía datos de los 10 canales adecuados para este fin por medio del puerto USB.

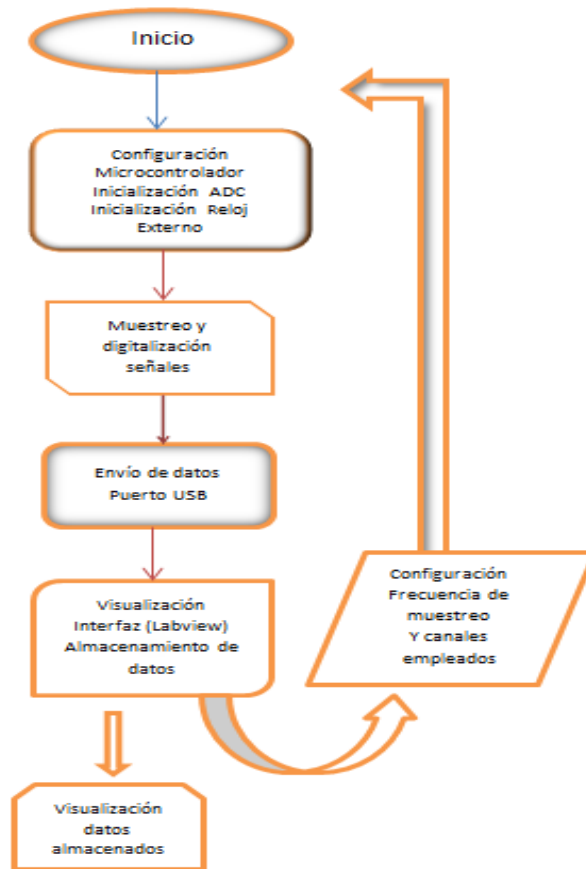
El usuario por medio de la interfaz podrá enviarle la orden al microcontrolador sobre qué canales del *ADC* desea que operen, así como también se podrá contar con la opción de seleccionar diferentes tipos de frecuencia (Fig. 3).

Figura 3: Interfaz Panel de control envío de datos.



Para iniciar el programa del microcontrolador bastará con conectarlo por medio del puerto USB del computador. Si se ha instalado previamente el software correspondiente, este lo detectará y lo pondrá en funcionamiento.

Figura 4: Diagrama de flujo tarjeta adquisición de datos.



El microcontrolador cuenta con dos (2) *switch*. El primero consiste un “reset” general de la tarjeta con el cual se reinicia el código del microcontrolador, abandonando el proceso en el que se encuentre y conectándose de nuevo al computador. El *switch* #2 permite al usuario elegir el momento en el cual desea empezar a transmitir datos desde el microcontrolador, esto con el fin de que no haya pérdida de datos o información, además esto permitirá al usuario adecuar la interfaz gráfica a los requerimientos que se desean en la prueba.

En cualquier momento de la prueba el usuario puede cambiar la frecuencia de muestreo (Fig. 3) y con esto conseguir en una misma prueba muestreos a diferente frecuencia sobre el mismo paciente.

3.2. CASO USB CLASE BULK

Para programar la tarjeta de adquisición de datos clase *BULK* se deben tener en cuenta algunos conceptos:

Transferencias: Una transferencia se puede definir como el conjunto global de los datos que forman una comunicación USB, una transferencia está formada a su vez por una o varias transacciones que a su vez están formadas por diferentes paquetes de datos que contienen las tramas de una comunicación USB.

No existe un formato único de transferencia, la especificación USB permite cuatro tipos de transferencias:

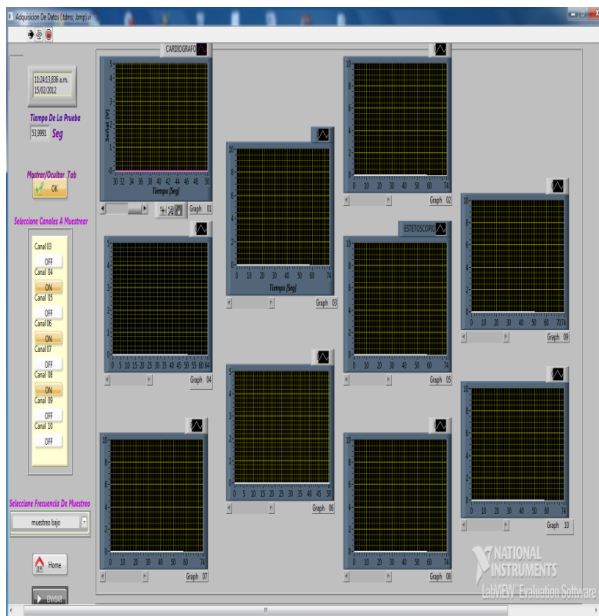
- Control: se utiliza para configurar y enviar comandos, por ejemplo en la enumeración del dispositivo.
- Bulk (masivas): se utilizan cuando se precisa una transferencia de datos grande, es el tipo más rápido de transferencia, sin embargo no hay garantía de que los datos se transmitan en un tiempo determinado (no garantizada la latencia). Sí hay verificación de que los datos se han transmitido con éxito ya que dispone de sistema de corrección de errores (*CRC*).
- Isócronas: Es usada en dispositivos que transmiten señales de audio y de vídeo en tiempo real. Se garantiza una tasa de velocidad de transmisión determinada (latencia asegurada).
- interrupción: latencia asegurada y verificación de que los datos se han transmitido con éxito, Se utiliza en dispositivos que no requieran mucho ancho de banda.

Cualquier dispositivo de comunicación USB usa al menos un tipo de transferencia, por ejemplo, la de control es utilizada siempre por todos los dispositivos en el proceso de enumeración (Cada vez que se conecte la tarjeta al sistema de cómputo).

4. DISEÑO E IMPLEMENTACIÓN INTERFAZ GRÁFICA

4.1. CASO USB CLASE CDC

Figura 5: Interfaz de usuario Adquisición de datos.



La interfaz gráfica de usuario, como su nombre lo indica, es el medio por el cual el usuario podrá interactuar con los componentes de la tarjeta de adquisición de datos. Esta comunicación se produce por medio del computador a través de puerto USB. El computador se comunica con el microcontrolador que a su vez ejecuta las tareas de acuerdo a los valores seleccionados por el usuario.

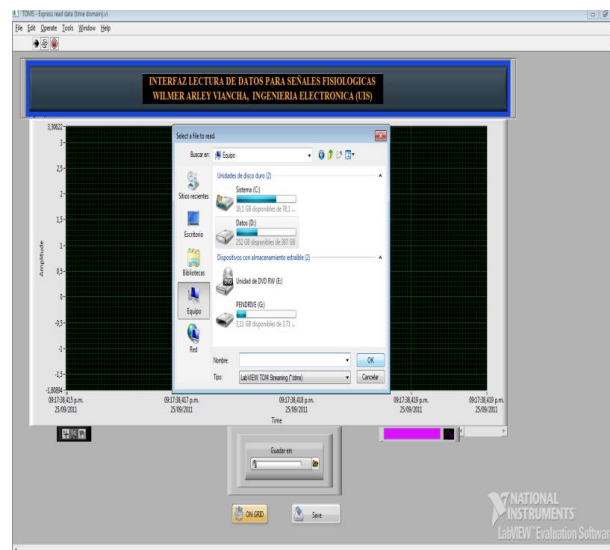
Para la adquisición y visualización de las señales fisiológicas se implementó una interfaz gráfica [Fig. 5], la cual se desarrolló con el software de *National Instrument* con el programa Labview.

Labview es un programa de programación que trabaja en lenguaje G, en otras palabras, emplea lenguaje gráfico y por medio de su entorno de programación por bloques permite rápidamente implementar aplicaciones para adquisición y visualización de datos.

La interfaz de visualización constará de dos componentes o programas. El primero se diseñó para observar la adquisición de las señales en tiempo real y almacenamiento de los mismos [Fig. 5]; El segundo se utiliza con el fin de visualizar los datos almacenados en el computador y los cuales se encuentran en un archivo con extensión *.tdms* [Fig. 6].

La interfaz de visualización de adquisición de datos consta de 11 pantallas para la visualización de las señales obtenidas. Una de las pantallas de visualización muestra todas las gráficas obtenidas y las superpondrá en el mismo eje; las otras diez muestran por separado cada una de las señales ingresadas [Fig. 5].

Figura 6: Interfaz gráfica Lectura de datos (Tipo de archivos *.tdms*)



Cuenta, a su vez, con una barra de control que se comunica con el microcontrolador y le enviará datos de frecuencia de muestreo y los canales de la tarjeta de adquisición de datos que se deseen habilitar de acuerdo al tipo de prueba que se esté realizando [Fig. 3].

La segunda fase de esta primera interfaz se encarga de almacenar los datos obtenidos, para dicho propósito. Esta interfaz guarda los datos ciclo a ciclo con el fin de evitar perder alguno. El usuario tendrá la posibilidad de elegir si desea guardar o no los datos al iniciar la sesión de operación, si es así, podrá elegir el nombre y ubicación del archivo, por defecto esta interfaz los guardara en el escritorio bajo el nombre de “**datos almacenados**” en extensión (***.tdms**). Esta clase de archivos se podrá visualizar con la interfaz diseñada con el segundo programa [Fig. 6], o si se desea hacerle un procesamiento a los datos se podrá abrir con el programa “**Microsoft Excel**” (desde la versión 2007 y posteriores).

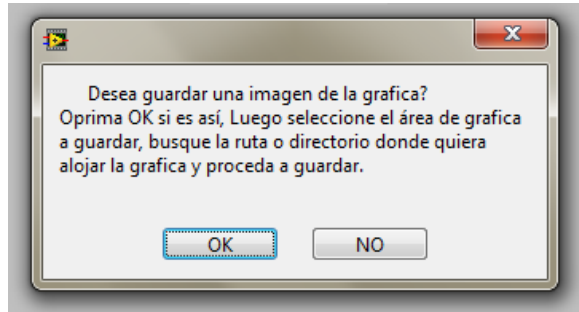
La razón por la que se guarda en este tipo de archivo será el permitir al usuario manipular y acceder a los datos obtenidos fácilmente sin necesidad de poseer un alto conocimiento de manipulación de datos o programación, ya que este programa hace parte de los programas básicos y esenciales bajo los cuales se inicia en el mundo informático. Sumado a esto, está el hecho de que Excel es una herramienta que posee y soporta muchas herramientas de cálculo para la manipulación matemática de los datos y/o su posterior exportación hacia otro software que el usuario desee utilizar.

Dentro de las opciones disponibles para esta interfaz grafica se pueden encontrar las siguientes:

* Guardar imagen (Opcional): permite al usuario la posibilidad de obtener una imagen en formato (*.bmp*) que puede ser abierto por cualquier visor para imágenes [Fig. 7]. En el modo adquisición de datos esta opción esta disponible

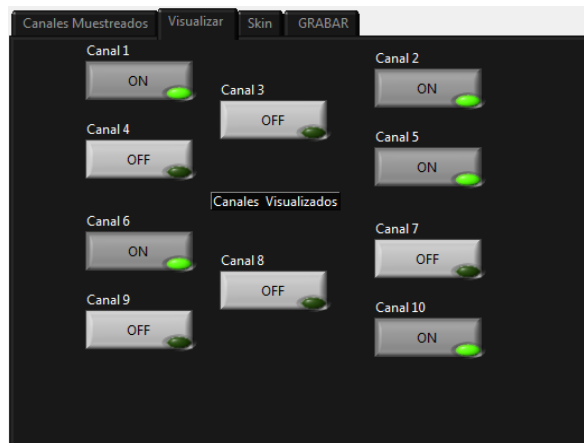
solamente para la gráfica que contiene la información de todas las gráficas (gráfica de datos superpuestos).

Figura 7: Mensaje: Ventana Grafica: ¿Desea Guardar imagen?



* *On/Off* Visualización de gráficas: (Interfaz adquisición de datos) permite al usuario cuales gráficas ver en la interfaz y cuales no [Fig. 8]. Esta opción con el fin de despejar la pantalla principal y permitir observar las gráficas más relevantes. Con esta opción no se habilitarán o deshabilitarán canales de muestreo de señales en el *ADC*. Esta función no impedirá que se guarde información de las señales que no se estén visualizando en el momento, si desea no guardar información sobre un canal este debe ser desactivado [ver Fig. 3].

Figura 8: Visualización de gráficas.



* Transparencia de la interfaz (*Skin*): permite al usuario ajustar el contraste de la interfaz.

* Botón *Home*: ofrece un atajo al usuario para saber cuál es el puerto que le asigno el computador a la tarjeta de adquisición de datos.

* Tiempo estimado de la prueba: permite al usuario saber el tiempo de duración que se lleva en la prueba.

4.2. CASO USB CLASE BULK

Para este tipo de comunicación se optó por trabajar con una frecuencia de muestreo de 1 [KHz], esto con el propósito de tener un procesamiento de señales con una frecuencia máxima de 500 [Hz]. Consecuentemente se capturarán señales fisiológicas de baja frecuencia, como las requeridas para esto proyecto, y se la posibilidad de capturar señales de una frecuencia aún mayor. Para la comunicación descrita, se diseñaron dos (2) Interfaz diferentes para que el usuario escoja la de su preferencia, de acuerdo al tipo de prueba que realice y la forma como quiera obtener los datos.

Estas aplicaciones cuentan con un indicador que le muestra al usuario si la tarjeta de adquisición de datos se encuentra conectada y si es reconocida por el sistema de cómputo. A su vez incluye las funcionalidades básicas presentadas con la comunicación USB clase *CDC*, como por ejemplo la duración de la prueba ó la opción de guardar imagen de las señales adquiridas en formato (*.bmp*).

En este orden, y teniendo en cuenta el volumen de datos recibidos, la interfaz le da opción al usuario de guardar o no la información obtenida por el sistema de adquisición de datos, ya que el tamaño del archivo generado va a ser de mayor tamaño en comparación con uno generado por USB clase *CDC*.

Sumado a lo anterior, esta interfaz cuenta con dos (2) tipos de gráficas diferentes, de acuerdo a las características de presentación de cada una de los modos de

visualización. Lo anterior, fue pensando para la comodidad del usuario. Las gráficas del tipo *Graph mode* se asemejan, en apariencia, al mostrado por un osciloscopio y las del tipo *Waveform mode* muestra una gráfica similar a las de un monitor utilizado para la obtención de signos vitales.

Nota:

Las aplicaciones expuestas anteriormente están diseñadas para trabajar sobre sistemas operativo *Windows*, para versiones *XP, Vista, Seven, 8*, con soporte en plataformas de 32 y 64 bits. A los instaladores anteriores, se adjuntan manuales de usuario, en los que se dan especificaciones de uso tanto, para hardware como para el uso de la interfaz de usuario.

5. Pruebas y Funcionamiento del sistema de adquisición de datos.

Esta sección expondrá los métodos bajo los cuales se verificó el funcionamiento de la tarjeta de adquisición de datos y la visualización de los mismos. Se describirán lo más detallado posible con el fin de transmitir una idea clara y concreta de lo realizado con el fin de comprobar el funcionamiento de la misma.

A continuación se describen los procesos realizados:

Teniendo los diez (10) canales habilitados en la tarjeta de adquisición de datos, para la recepción de señales de baja frecuencia, se procede a realizar las pruebas de máxima velocidad de transmisión de datos, así como también la verificación de datos enviados a la interfaz de usuario.

Para encontrar la máxima velocidad de transmisión de datos se programó el microcontrolador con valores constantes, los cuales simulaban las 10 señales adquiridas, de acuerdo al tipo de comunicación programado se logró llegar a los siguientes resultados:

Para la comunicación USB tipo *CDC* se logró llegar a transmitir un máximo de 160 Bits cada 4 [ms], por tanto, se logra transmitir un máximo de 40.000

[Bits/segundo]. Lo anterior representa una velocidad máxima de 5.000 [Bytes/seg], por lo que se obtendrá una velocidad de muestreo de 250 [Hz]. Basándose en la teoría de *Nyquist*, la máxima señal que se podría trabajar bajo USB clase *CDC* para este caso sería una señal con una frecuencia máxima 125[Hz].

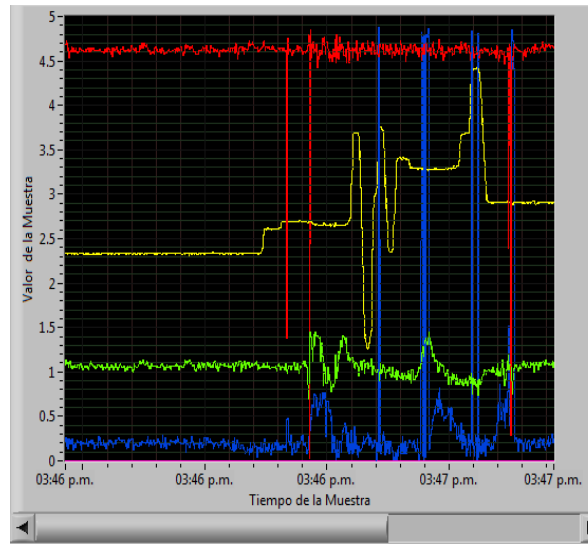
Para la comunicación USB tipo *BULK* se puede lograr una velocidad máxima de 12 [MB/s]. Lo anterior es válido si, y solo si, el bus de datos del puerto USB del sistema de cómputo, al cual se ha conectado la tarjeta, está disponible solo para la tarjeta de adquisición de datos, puesto que al compartir el ancho de banda del mismo con otro dispositivo que trabaje con protocolo USB, este se repartirá para los dispositivos que se encuentren conectados. Para cumplir con el propósito por el cual se diseñó la tarjeta de adquisición de datos, se realizaron pruebas enviando 160 Bits cada milisegundo, logrando 160.000 [Bits/segundo] o lo equivalente a una velocidad de 20 [KB/s], con esto se permite muestrear señales que tengan una frecuencia máxima de 500 [Hz], ya que la velocidad de muestreo es de 1 [KHz].

NOTA: La prueba descrita anteriormente para la comunicación USB tipo *CDC* se realizó en un computador que cuenta con un procesador *Intel core i5*. La aclaración anterior se refiere al tipo de comunicación empleado, ya que al realizarla con un procesador de menos poder se obtienen un retraso de hasta 5 segundos del dato real frente al visualizado en pantalla en ese instante. La situación descrita es resultado de realizar simultáneamente la adquisición de datos junto con otras tareas que ejecutaba el procesador (navegar en internet, escuchar música, tener programas que requieren la atención del procesador, entre otras)

La prueba que se describe a continuación se realizó utilizando generadores de señales junto con sensores y potenciómetros ubicados en una tarjeta de desarrollo utilizada en prácticas de laboratorio, que hacen parte de los

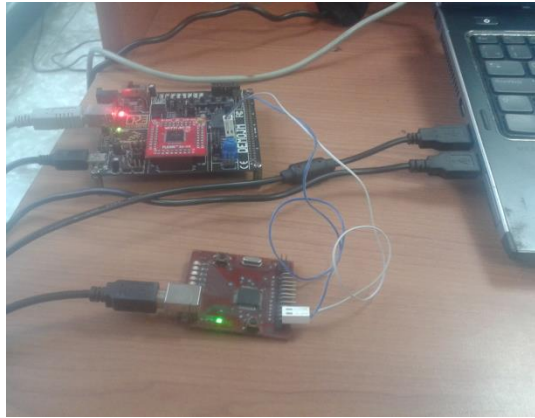
laboratorios de la escuela de ingenierías eléctrica, electrónica y telecomunicaciones de la universidad industrial de Santander.

Figura 9: Recepción de señales en tarjeta de adquisición de datos, señales provenientes de sensores en placa de desarrollo.



En dicha prueba se corroboró la veracidad de la información mostrada en pantalla, en comparación con las señales tomadas de los generadores de señales. Para realizar dicha prueba se tomaron dos (2) señales por medio de los generadores de señales. También se ingresó una señal por medio de un circuito serie, el cual consta de una batería de 6 [V], una resistencia de valor constante de 10 [K] y una resistencia de valor variable (Potenciómetro) de 100 [K], las otras señales fueron simuladas internamente por el microcontrolador. El procedimiento realizado para la prueba anterior fue el siguiente:

Figura 10: Visualización de señales tomadas por Generador de señales.

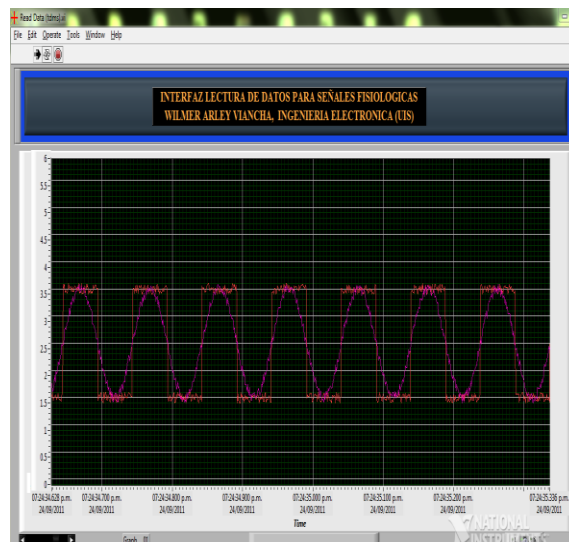


Respecto a las señales tomadas por los generadores de señales, se trabajó con frecuencias de hasta 125 [Hz] (senoidal, cuadrada).

Para la señal tomada del circuito serie, la salida se fijó en la resistencia variable, Las precauciones tomadas al realizar la prueba, así como también cualquier adquisición, de datos fueron las siguientes:

La amplitud de la señal a digitalizar no debe superar el valor de alimentación del microcontrolador + 0.3 [V]. Para el caso particular de esta tarjeta, el valor corresponde a 5.3 [V], pues al ingresar un valor mayor, dañaría permanente el microcontrolador que hace parte de la misma.

Figura 11: Visualización de señales provenientes de sensores (Acelerómetro de tres ejes (X, Y, Z), potenciómetro).



- El sistema de adquisición de datos no puede digitalizar señales con valores negativos (-), en caso tal de contar con valores negativos en la señal a adquirir, se recomienda agregar un nivel de *offset* (valor conocido) y posteriormente hacer la respectiva corrección por *software* del valor agregado, así como una posible solución para valores mayores a 5.3 [V] se recomienda el uso de amplificadores con ganancia menor a la unitaria, y luego hacer la respectiva corrección por *software*.
- Esta tarjeta fue diseñada para la digitalización de señales de baja frecuencia, dichas señales provienen de dispositivos diseñados y construidos en base a al proyecto macro del profesor Salvador Pacheco, quien figura como codirector del presente proyecto, y en base a los requerimientos de dicho proyecto se diseñó y construyó la misma, por tanto no se tendrá ningún riesgo a la hora de leer las señales analógicas provenientes de dichos dispositivos, ya que estos fueron diseñados bajo la especificación requerida en esta tarjeta. Para el uso de esta tarjeta en otra tarea ajena al propósito bajo el cual fue construida, se recomienda tener cuidado con interferencia de señales presentes en el ambiente, así como la sobre tensión en las señales a leer.

Para un uso adecuado y un conocimiento más profundo sobre el uso tanto de esta tarjeta como del *software* de visualización de datos, se recomienda al lector remitirse al documento “Manuales de operación”.

Para la entrega final de este proyecto no hay ningún valor de corrección de *offset* ni de amplitud en las señales visualizadas, lo anterior debido a que el alcance del proyecto no incluye la manipulación de las señales recibidas, sin embargo, se

dejará el código para libre acceso con el fin de hacer los ajustes correspondientes según sea el caso.

En cuanto a los resultados arrojados por la prueba realizada, se hicieron diferentes combinaciones como las variaciones de amplitud de la señal, las variaciones de frecuencia de señal y las variaciones del canal por el cual se adquiere la señal. Se corroboró el correcto funcionamiento de la tarjeta de adquisición de datos, esto se verificó visualizando los datos por medio de la interfaz gráfica y comparándolos con los valores de las señales generadas por los generadores de señal.

6. CONCLUSIONES Y RECOMENDACIONES

La comunicación USB clase *Bulk* ofrece una velocidad de transmisión de datos mayor, en comparación al protocolo de comunicación USB clase *CDC*, por lo tanto, este tipo de protocolo es apropiado cuando el volumen de datos y la velocidad de transmisión es un factor crítico.

La simulación de la comunicación *USB-CDC* requiere mas capacidades del sistema de computo bajo el cual se adquieran los datos, lo anterior se presenta debido a los drivers con los cuales se maneja esta comunicación, pues estos están diseñados para trabajar con las especificaciones propias del protocolo de comunicación *RS232*, como consecuencia de lo anterior la comunicación *USB-CDC* puede presentar atrasos en la visualización de datos, perdida de información o lo que puede llegar a ser aun peor, recibir información errónea, esto se presenta cuando se trata de llevar la comunicación a una velocidad igual o mayor a 320.000 [bit/seg] y no se cuente con un procesador de ultima generación⁷.

Las herramientas que contiene *Labview* para la adquisición de señales son óptimas para adquirir datos en tiempo real, sin embargo, se debe tener en cuenta que la maquina de visualización (computador) no es de propósito específico, siendo esto una limitante ya que la eficiencia, eficacia y velocidad del procesamiento de los datos depende de la configuración del *hardware* que posea el equipo de computo.

El contar con una tarjeta de adquisición de datos que permita realizar un estudio de diferentes señales fisiológicas simultáneamente, permitirá obtener resultados más exactos sobre el estado de salud del paciente, de esta manera, se tendrá

⁷ La expresión “última generación” se refiere en este texto a procesadores Intel desde Core i5 o sus equivalentes en otros fabricantes de procesadores para computadores.

conocimiento de posibles enfermedades que puedan surgir y con esto iniciar un tratamiento para mejorar la calidad de vida del paciente o evitar riesgos a futuro.

Por otro lado, esta tarjeta ayudará a los estudiantes de la facultad de salud en su formación académica, pues les permitirá realizar prácticas y diferentes exámenes sobre un mismo paciente; así mismo, ayudara a fortalecer los laboratorios de la facultad de salud mejorando la calidad de la educación que presta la Universidad Industrial de Santander (UIS). Como resultado de esto, formará profesionales que ayuden en la problemática de la región y ofrezcan soluciones al mundo en el sector de la salud.

La visualización de los datos en "tiempo real" (retardo de visualización menor a 1 segundo) permite observar e identificar en qué parte del proceso o examen ocurren fallas en el paciente; esto se logra sometiénolo a diferentes estímulos, pruebas de estrés u otros métodos, encontrando así el momento y la causa que puede producir el anormal funcionamiento del órgano(s) en estudio.

Para versiones mejoradas de este prototipo se recomienda la transmisión de datos inalámbricamente, ya que esto facilitará el uso del dispositivo en otro tipo de exámenes, en los cuales la movilidad y lo problemático de los cables son un factor limitante.

Se recomienda diseñar interfaz gráfica con lenguaje java u otro software libre de esta manera se podrán distribuir y comercializar copias de esta versión a diferentes personas u organizaciones, así mismo conectarla a una base de datos con el fin de llevar un historial en cada paciente, en el cual se pueda acceder a la cuenta de usuario de cada persona y ver todos los registros y notas asociadas a este.

BIBLIOGRAFIA

AMBARDAR, Ashok. Procesamiento de señales analógicas y digitales. Michigan: Michigan Technological University, 2002. 814 p.

An Approach to Continuous Optimization by the Intelligent Water Drops Algorithm. En: Procedia - Social and Behavioral Sciences. September, 2010. vol. 32

ARIAS, Juan y MOGOLLÓN, Moisés. Algoritmo de optimizaciones gotas de agua virtuales inteligentes aplicado a la planeación de ruta óptima de un robot móvil. Trabajo de grado para optar por el título de Ingeniero Electrónico. Bucaramanga: Universidad Industrial de Santander. Facultad de Ingeniería Físico-Mecánicas. 2013. 63 p.

CENTENO, Diego y MORENO, Fabio. Análisis comparativo del desempeño de técnicas de control conmutado implementadas en dispositivos programables. Trabajo de grado para optar por el título de Ingeniero Electrónico. Bucaramanga: Universidad Industrial de Santander. Facultad de Ingeniería Físico-Mecánicas. 2013. 92 p.

CHERNIZ ANALIA, Cian Leandro M. Sistema de adquisición multicanal para el registro de señales electrofisiológicas. Argentina: Universidad Nacional de Entre Ríos, 2008. 45 p.

COLLANTE, Andrés. Métodos numéricos para ecuaciones diferenciales ordinarias con MATLAB. Trabajo de grado. Bellavista: Universidad Nacional del Callao. Facultad de Ingeniería Mecánica Energía, 2011. 70 p.

CONG, Shuang; LI, Guodong y FENG, Xianyong. Parameters identification of nonlinear dc motor model using compound evolution algorithms. En: World Congress on Engineering. Londres: Junio-Julio, 2010.

DENG, Xiuqin. System identification based on particle swarm optimization algorithm. En: Computational Intelligence and Security. Beijing: noviembre, 2012.

DUARTE, César y QUIROGA, Jabid. Algoritmo PSO para identificación de parámetros en un motor DC. Rev.fac.ing.univ. Antioquia [en línea] . http://www.scielo.org.co/scielo.php?script=sci_arttext&pid=S0120-62302010000500012&lng=es&nrm=iso. ISSN 0120-6230

FESSLER, Jeffrey y ARBOR, Ann. On transformations of random vectors: communications & signal processing laboratory. Michigan: University of Michigan. 1998. 75 p.

GARRIDO, Santiago. Identificación, estimación y control de sistemas no-lineales mediante RGO. Tesis doctoral. Leganés: Universidad Carlos III de Madrid. Departamento de Ingeniería Eléctrica, Electrónica y Automática, 1999. 161 p.

KAMKAR, Iman; AKBARZADEH-T, Mohammad-R y YAGHOOBI, Mahdi. Intelligent Water Drops a New Optimization Algorithm for Solving the Vehicle Routing Problem. En: IEEE International Conference on Systems Man and Cybernetics (SMC). Turquía: octubre, 2010. 400 p

KUMON, Toshiro; IWASAKI, Makoto; SUZUKI, Tatsuya; HASHIYAMA, Tomonori; MATSUI, Nobuyuki y OKUMA, Shigeru. Nonlinear System Identification Using Genetic Algorithm. En: 26th Annual Conference of the IEEE Industrial Electronics Society. Japón: octubre, 2010. 200 p.

LIU, Guangjun; XIAOPING, Xu, y FENG, Wang. Identification of a Kind of Nonlinear System. En: Seventh International Conference on Natural Computation. Shanghai, 2011. 100 p.

López Pérez, Eric. Protocolo USB (UNIVERSAL SERIAL BUS). [en línea]. <http://www.i-micro.com/pdf/articulos/usb.pdf>

Maldonado Pedro y Vivaldi Ennio, Computadores en investigación biomédica II: Control experimental, adquisición y almacenamiento de datos. En: Revista médica de Chile. Septiembre 2001, Vol. 129, No.9. [en línea]. http://www.scielo.cl/scielo.php?script=sci_arttext&pid=S0034-98872001000900017

MOLGA, Marcin y SMUTNICKI, Czesław. Test Functions for Optimization Needs. Test Functions for Optimization Needs. The Bioinformatics Laboratory. 2005. 43 p.

National Instruments, Comunicación Serial: Conceptos Generales, [en línea]. <http://digital.ni.com/public.nsf/allkb/039001258CEF8FB686256E0F005888D1>

PALLÁS ARENY, Ramón. Adquisición y distribución de señales. España: Universidad politécnica de Catalunya, 2005. 426 pg.

Problem Solving by Intelligent Water Drops. En: IEEE Congress on Evolutionary Computation. Singapore: septiembre, 2007. 333p

SHAH-HOSSEINI, Hamed. Optimization with the Nature-Inspired Intelligent Water Drops Algorithm. En: PINHEIRO DOS SANTOS W. Evolutionary Computation. 2009. 297p.

USB.ORG. Normatividad protocolo comunicación USB [en línea].
<http://www.usb.org/developers/docs/>

XU, Xiaoping; WANG, Feng y QIAN, Fucai. Study on Method of Nonlinear System Identification. En: International Conference on Intelligent Computation Technology and Automation. China: September-October, 2011, vol. 45.