

Desarrollo de un instrumento virtual con tecnología USB

Development of a virtual instrument with USB technology

RICARDO PIRAJÁN CANTILLO

Ingeniero electrónico y especialista en Teleinformática, Universidad Distrital Francisco José de Caldas. Profesor de tiempo completo adscrito a la Facultad Tecnológica de esta Universidad. Miembro del Comité de Autoevaluación y Acreditación de la Facultad Tecnológica y director del Grupo de Investigación en Instrumentación Virtual “Integra”, de la misma Facultad.

rpirajanc@udistrital.edu.co

SANDRA MILENA CÁRDENAS B.

Tecnóloga electrónica Universidad Distrital Francisco José de Caldas, auxiliar del laboratorio de Software Aplicado de Tecnología en Electrónica. Integrante del Grupo de Investigación en Instrumentación Virtual “Integra”.

smcardenasb@msn.com

JUAN DAVID FONSECA MONTOYA

Tecnólogo electrónico, Universidad Distrital Francisco José de Caldas Integrante del Grupo de Investigación en Instrumentación Virtual “Integra”.

jdfonsecam@ieee.org

Fecha de recepción: abril 15 de 2004.

Clasificación del artículo: Investigación.

Fecha de aceptación: junio 30 de 2004

Palabras clave: osciloscopio digital, instrumentación virtual, USB, dispositivo de interfaz humana.

Key words: digital oscilloscope, virtual instrumentation, USB, human interface device.

RESUMEN

En este artículo se presentan los resultados obtenidos por el Grupo de Investigación en Instrumentación Virtual “Integra”, en el desarrollo de un osciloscopio virtual utilizando como interfaz de comunicaciones el bus USB. Se hace especial énfasis en el módulo de comunicaciones, ya que éste ha sido uno de los aportes principales de este proyecto. Para el desarrollo del programa se utilizó Visual Basic.

ABSTRACT

This paper shows the results obtained by the research group in virtual instrumentation “INTEGRA” in the development of a virtual oscilloscope using USB as communication interface. A special emphasis is made in the communications module since this has been one of the main contributions for this project. For the development of the program Visual Basic was used.

1. Introducción

La instrumentación electrónica puede ser definida como aquel campo especializado en realizar mediciones de magnitudes físicas, eléctricas o no. Para esto se desarrollan dispositivos electrónicos que van desde registradores simples hasta sistemas sofisticados de adquisición de datos vinculados a computadores. Las tareas de supervisión y control de procesos industriales se han optimizado gracias al desarrollo tecnológico del área de instrumentación, logrando equipos de medida con mayores prestaciones y confiabilidad.

Los computadores desempeñan un papel importante en el mundo de hoy, y constituyen una herramienta fundamental en los desarrollos presentes y futuros de la instrumentación no convencional, definiendo un campo relativamente nuevo en el medio colombiano: la instrumentación virtual. El instrumento virtual está constituido de una tarjeta de captura de señales y software para su análisis y visualización.

Entendiendo la necesidad de apropiar los conocimientos sobre los últimos desarrollos tecnológicos a nivel mundial, el grupo de investigación “Integra” ha centrado su interés en el bus de comunicaciones USB (*Universal Serial Bus*), para utilizarlo como medio de comunicación en aplicaciones de instrumentación virtual, entre ellas en el proyecto “Osciloscopio UD”.

2. Osciloscopio UD

El uso de los computadoras personales para la emulación de instrumentos virtuales (multímetros, analizadores de espectro, osciloscopios, etc.) ha obtenido un gran desarrollo, a tal punto que en la actualidad es común hablar de redes de instrumentación o sistemas de supervisión que simplifican el análisis electrónico, automatizando las operaciones que antes se hacían en forma manual.

El osciloscopio UD hace parte de los primeros desarrollos del Grupo de Investigación “Integra”. Las características principales de este instrumento son: dos canales, ancho de banda de 1 MHz, 10 Vp-p en entrada máxima, tasa de muestreo de 60 Ms/s (millones de muestras por segundo), impedancia de entrada de 1 Megaohmio, control de disparo (*trigger*), control de dispositivos internos mediante CPLD (dispositivo lógico programable complejo), comunicación USB con el PC, y una aplicación de software para el análisis y visualización de los datos. El instrumento consta de tres componentes: hardware, firmware (programa del microcontrolador) y software.

2.1 Descripción de los bloques funcionales del osciloscopio

En la figura 1 se muestran los principales bloques funcionales del osciloscopio UD; cada uno de ellos será descrito en las siguientes secciones.

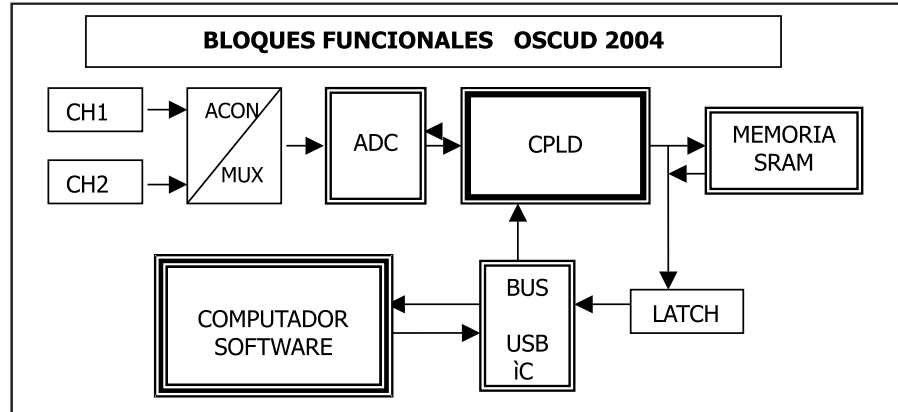


Figura 1. Bloques funcionales del osciloscopio

2.1.1. Bloque de acondicionamiento de señal y multiplexación de canales

La señal análoga adquirida por cada canal debe ser acondicionada para que se adecue a las características del convertor análogo - digital, garantizando que no se afecte por el sistema de medida mediante la utilización de amplificadores operacionales y arreglos RC (resistencia y condensadores).

La señal permanece dentro de una ventana de 4V, con un nivel de desplazamiento DC de 1V (*offset*), dado que con estas especificaciones funciona el convertor análogo a digital empleado en este diseño. La selección de los canales es obtenida con el integrado 74HCT4052, el cual es un multiplexor análogo de respuesta rápida que no presenta una atenuación significativa a la señal.

Con este integrado pueden medirse diversas señales con un único convertidor analógico-digital. Este convertidor obtiene una muestra de un canal, e inmediatamente después conmuta al siguiente canal de entrada, por lo que un sistema de adquisición sólo necesita un convertidor para muchos canales. Esto significa que la velocidad de muestreo de cada canal individual es inversamente proporcional al número de canales muestreados.

2.1.2. Bloque de conversión análogo a digital

La señal de entrada ha de ser muestreada de acuerdo con el criterio de Nyquist: “Para reconstruir cualquier señal continua a partir de una señal muestreada, la frecuencia de muestreo debe ser al menos el doble de la componente de frecuencia más alta de la señal en cuestión”; por tanto es importante que el convertidor analógico-digital pueda convertir la señal en bits en el menor tiempo posible. Otro parámetro clave de este convertidor es la resolución, que se puede definir como el número de bits que utiliza para representar una señal análoga.

El bloque de conversión análogo a digital consta básicamente del integrado HI5662 de la empresa Intersil. Este convertidor tiene como características principales una tasa de muestreo máxima de 60 Ms/s (megamuestras por segundo), dual (dos canales con conversión individual y un único reloj), niveles de referencia interna y resolución de 8 bits; se eligió entre varias referencias dado que cumple con los objetivos del diseño, entre ellos la velocidad de muestreo y su respectiva resolución, garantizando anchos de banda aceptables.

2.1.3. Bloque de control CPLD (Complex Programmable Logic Device)

En este bloque se encuentra el CPLD de ALTERA, referencia EPM7128SLC84-15 de la serie MAX7000. Sobre el CPLD se encuentra el control de los dispositivos y el bus de datos proveniente del convertidor análogo a digital. Además, tiene la decodificación y ejecución de instrucciones provenientes del software. El firmware está constituido por un bloque general compuesto de bloques funcionales (véase figura 2), programados en lenguaje VHDL y AHDL.

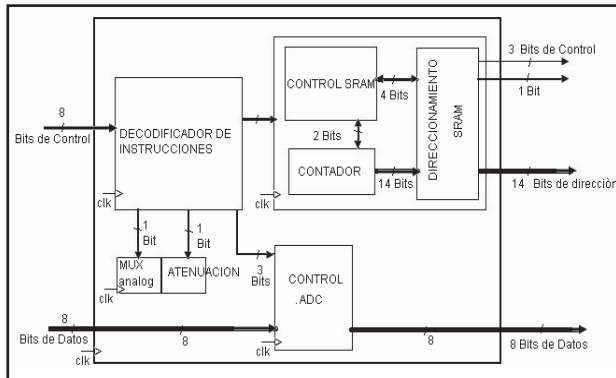


Figura 2. Bloque global y bloques funcionales

El CPLD de la empresa Altera se escogió como control lógico porque se acomoda a las especificaciones del diseño y se cuenta con el sistema de desarrollo (programador). Como opción se consideró un FPGA (arreglo de compuertas programables en campo), pero se descartó porque necesita ser reprogramado cada vez que se energiza.

En la figura 3 se muestra un fragmento del diagrama de flujo; en él se explica brevemente la función del CPLD en el control y la selección del canal.

2.1.4. Bloque de memoria

En él se guardan temporalmente los datos de 8 bits pro-

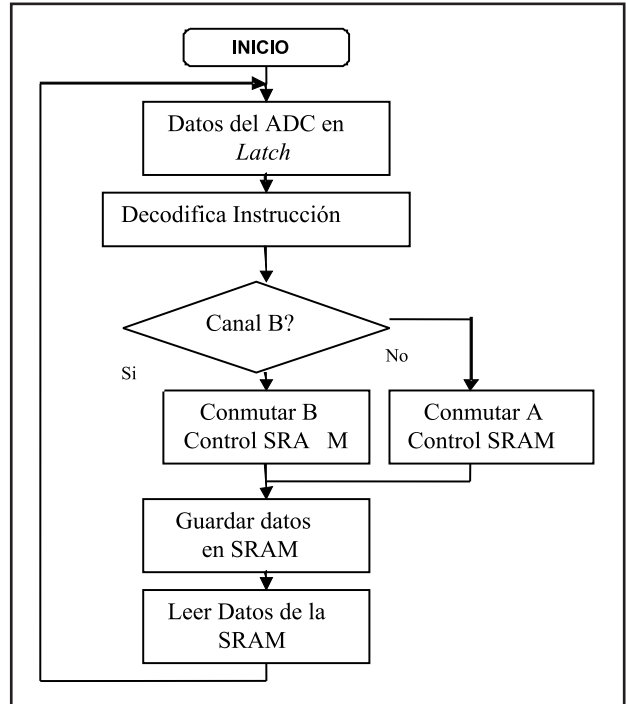


Figura 3. Diagrama de flujo CPLD

venientes del convertidor análogo a digital y controlados por el CPLD. Este bloque hace la función de *buffer*; el principal y único componente es una memoria SRAM de 32 k x 8 de la empresa Cypress, cuya principal virtud es su tiempo de acceso de 15 ns, acorde con las necesidades del diseño. La memoria es direccionada y controlada por el CPLD y es segmentada en dos; las primeras posiciones de memoria son para el canal 1, y las restantes para el canal 2.

2.1.5 Bloque de comunicación USB

En este bloque se genera la comunicación entre el computador y el CPLD por medio del bus USB; la comunicación se obtuvo empleando el microcontrolador de Microchip 16C765, el cual cuenta en su arquitectura con un módulo de comunicación USB; este tipo de comunicación permite conectar el osciloscopio al computador sin necesidad de destapar o adicionar hardware: sólo mediante un conector y un programa que se comunique con el controlador de dispositivos USB, instalado en el PC, y la aplicación del osciloscopio.

3. Bus Serial Universal (USB)

El primer avance en el diseño del osciloscopio virtual fue la comunicación USB; al respecto se trabajaron sus principales características, desde el nivel físico

(características eléctricas y físicas) hasta su nivel de aplicación (*firmware*).

3.1 Aspectos generales

La tecnología USB ha sido uno de los más importantes desarrollos en la interconexión de dispositivos a un computador. La facilidad de instalación y configuración de los periféricos es una de las principales ventajas de este puerto, dado que los dispositivos son diseñados bajo el esquema *plug and play*. Esta característica, sumada a su velocidad y alto rendimiento, son algunas de las razones por las cuales esta tecnología, relativamente nueva, ha sido desarrollada con especial interés en los últimos años (1996- 2004) (Axelson, 2001, 3).

La primera versión de USB fue la 0.9, presentada al mundo como un borrador en noviembre de 1995; en la versión 1.0, presentada el 15 de enero de 1996, se establecen dos tipos de conexión: a) una denominada *low speed* (de baja velocidad), con la cual se logra una velocidad de transmisión de 1,5 Mbps, desarrollada para periféricos que no requieren un gran ancho de banda, como ratones o mandos de juegos; b) la conexión *full speed* (de alta velocidad), que permite una velocidad de transmisión de 12 Mbps y está destinada a los dispositivos más rápidos como impresoras. Posteriormente, en el año 1998 fue presentada la versión 1.1.

El desarrollo del USB versión 1.0 es el resultado del trabajo de siete empresas líderes en tecnología, comunicaciones y desarrollo de computadores¹. En la actualidad, la versión 2.0 cuenta con una opción de velocidad adicional denominada *high speed* (alta velocidad), con una rata de 480 Mbps, superando por un amplio margen las prestaciones del puerto serial y paralelo, y a su principal competidor, *firewire* (400 Mbps) (USB-IF, 2004).

3.2 Especificaciones técnicas

3.2.1 Cable y conectores

La conexión física con el computador se establece mediante un cable de cuatro hilos; dos son de alimentación (5 voltios y tierra) y dos de comunicación (D+ y D-), utilizando comunicación balanceada (diferencial). Uno (1.0) o cero (0.0) son los resultados de la diferencia de las líneas indicada en la tabla 1; la distribución de pines se muestra en la figura 4 (Anderson, 2003,69-74).

¹ Compaq, Digital Equipment Corporation, IBM PC Company, Intel, Microsoft, NEC y Northern Telecom

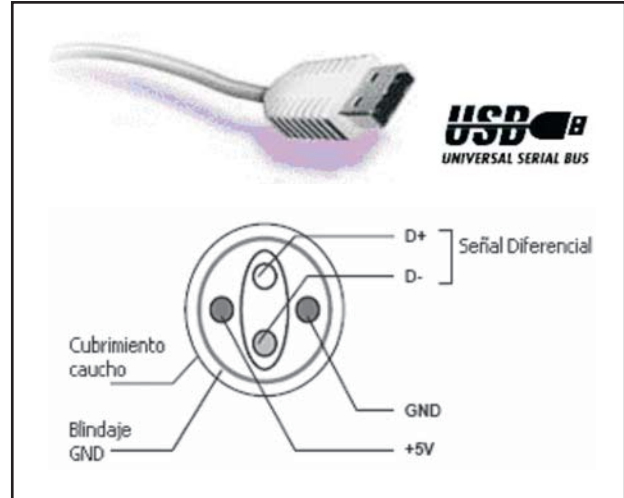


Figura 4. Cable USB. Pines de conexión y logotipo

Tabla 1. Niveles de la señal

	Origen	Recepción
"0" Diferencial	D+ > VOH (mín) y D- < VOL (máx)	D- - (D-) > 200mV y D+ > VIH (mín)
"1" Diferencial	D- > VOH (mín) y D+ < VOL (máx)	(D-) - (D+) > 200mV y D- > VIH (mín)
	D+ y D- < VOL (máx)	D+ y D- < VIL (máx)

Los cables tienen dos tipos de conectores estándar, A y B, para establecer la conexión física con el computador; ellos cuentan con sus terminales hembra y macho; las primeras se encuentran ubicadas en los dispositivos (CPU o impresora); las segundas, en el cable del dispositivo periférico (*mouse* o teclado). Así, los conectores tipo A son utilizados para la conexión con el computador (véase figura 5a), y el tipo B para la conexión al dispositivo (figura 5b). En la tabla 2 se indican los nombres de los pines con su respectiva función.

Tabla 2. Descripción de pines

Pin	Nombre	Descripción
1	VBUS	+5 VDC
2	D-	Dato -
3	D+	Dato +
4	GND	Tierra

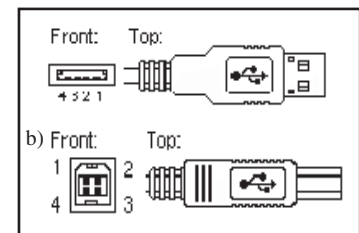


Figura 5. Conectores USB. a) tipo A; b) tipo B

Este tipo de cable permite alimentar dispositivos externos, clasificables de acuerdo con su consumo de corriente en bajo (hasta 100 mA) y alto (hasta 500 mA). Para dispositivos de más de 500 mA será necesario alimentación externa; el nivel de voltaje máximo es de 5V ± 10% (Hyde, 2001,15).

La transmisión de datos utiliza el código de línea NRZI (no retorno a cero invertido); esta codificación cambia el nivel de la señal cada que vez que se presenta un cero pasando de V+ a V-, o de V- a V+, de acuerdo con el caso. En la figura 6 se muestra un ejemplo de codificación mediante NRZI; asimismo, la figura 7 presenta el diagrama de bloques de las secciones de transmisión y recepción con USB.

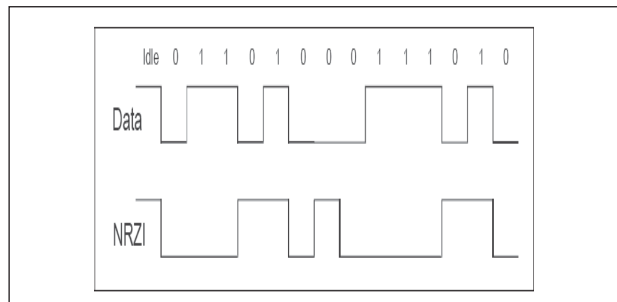


Figura 6. NRZI (no retorno a cero invertido)

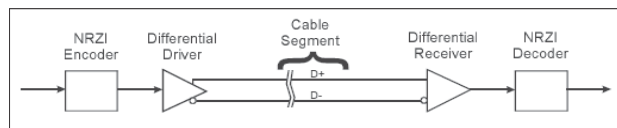


Figura 7. Bloques de TX y RX con USB

Cuando hay más de un dispositivo conectado a un puerto mediante un *hub*, el ancho de banda es dividido en tramas de un milisegundo, para 12 Mbps, y 125 μs para *high speed*, mediante multiplexación por división en el tiempo (*time división multiplex*). Esto permite que todos los dispositivos se comuniquen con la misma velocidad de transmisión.

3.3 Topología del bus USB

El bus cuenta con dos tipos principales de elementos: *host* (servidor) y dispositivos; estos elementos se interconectan mediante una topología en estrella escalonada (*tiered star*), como se muestra en la figura 8. Para cada sistema se cuenta únicamente con un *host*, del cual se pueden desprender hasta 127 dispositivos, mediante la utilización de concentradores (*hubs*). En un PC pueden

encontrarse uno o más *host*², que utilizan una solicitud de interrupción (IRQ), sin importar cuántos dispositivos estén conectados directamente a un *hub* raíz o externo; de esta manera evita las colisiones y limitantes de hardware (Ming, 1997, 21).

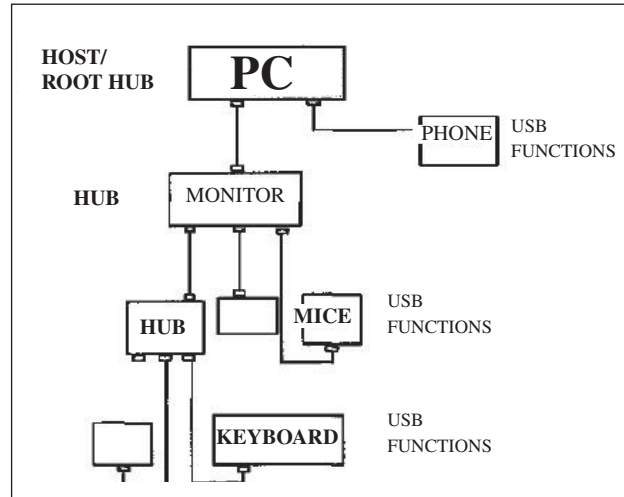


Figura 8. Topología USB

3.4 Flujo de datos

El control de la comunicación es realizado por el *host*, y ningún dispositivo puede transmitir si éste no lo solicita (*polling*); los dispositivos deben esperar un turno para transmitir y no pueden generar ningún mensaje para indicar que desean comunicarse. El *host* USB genera un listado de dispositivos mediante el cual organiza el sondeo y pasa por cada uno de ellos para activar su transmisión; este método de acceso es del tipo *token bus*, que otorga igual importancia a cada dispositivo; el *host* tiene el control maestro (Fonseca y Fino, 2003, 121).

3.5 Tipos de transferencia

En la tecnología USB existen cuatro tipos de transferencia: de control, isócrona, de interrupción y *bulk* (por volumen). El detalle puede verse en la tabla 3.

- *Transferencia de control*: es bidireccional y se utiliza para realizar comunicación de configuración, comandos o estado entre el *host* y los dispositivos; consta de dos o tres etapas: establecimiento (*setup*), transferencia de datos³ y estado (*status*).

² El número de puertos USB no debe confundirse con el número de *host*.

³ Que puede no existir.

Tabla 3. Tipos de transferencia

Tipo de transferencia	Control	Bulk	Interrupción	Asíncronicos
Uso típico	Configuración	Impresoras, escáner	Teclados, ratones	Audio
¿Requerido para establecer comunicación?	Sí	No	No	No
¿Presente en dispositivos de baja velocidad?	Sí	No	Sí	No
Velocidad de los datos bytes/milisegundos	15872	53248	424572	24576
Dirección de los datos	Entrada y salida	Entrada o salida	Entrada o salida (entrada solamente en versión 1.0)	Entrada o salida
Ancho de banda reservado	10 a low/full speed 20 a high speed	Ninguno	90 a low/full speed 80 a high speed	
Corrección de errores	Sí	Sí	Sí	No

- *Transferencia isócrona:* puede ser unidireccional o bidireccional; es usada para la transferencia de información cuando se requiere una tasa de velocidad constante y no se pueden tolerar errores; un ejemplo de aplicación es la transferencia de datos de voz tipo PCM de 64 kbps. El máximo paquete de transmisión isócrona es de 1023 bytes por trama USB de 1 ms; esto implica que la máxima transferencia de este tipo es de 8,148 Mbps.
- *Transferencia por interrupción:* es unidireccional y solamente de entrada hacia el *host*. Se emplea para transferencia de datos pequeños. Es de tipo *polling* o encuesta; un ejemplo es un teclado, el cual envía pequeños datos hacia el *host* cada vez que se presiona una tecla.
- *Transferencia bulk* (transferencia por volumen): puede ser uni o bidireccional. Se emplea para transmitir grandes volúmenes de datos cuando el tiempo no es crítico. Un ejemplo típico es la transferencia de datos de un escáner hacia el PC.

4. Implementación de la comunicación USB

Las comunicaciones USB del osciloscopio UD se basan en el microcontrolador PIC16C765 de Microchip, el cual posee un módulo de comunicación USB (*low speed*) y un módulo de conversión análogo a digital de ocho canales con resolución de 8 bits.

4.1 Características del microcontrolador PIC16C765 de Microchip

Este microcontrolador es miembro de la serie 16C5x. El código fuente es compatible con los 16C7X4, que

soportan 35 instrucciones (véase figura 9). Debido a que el microcontrolador soporta únicamente transferencia USB de baja velocidad, cuenta con seis *endpoint* y permite sólo dos tipos de comunicación: de control y de interrupción, en cualquier combinación de entrada y salida. Microchip provee de notas de aplicación de lenguaje ensamblador y lenguaje C para la enumeración y otras tareas USB. Para dispositivos de interfaz humana HID existe una nota de aplicación de un *mouse*, el cual sirve para adaptarlo a otras aplicaciones HID.

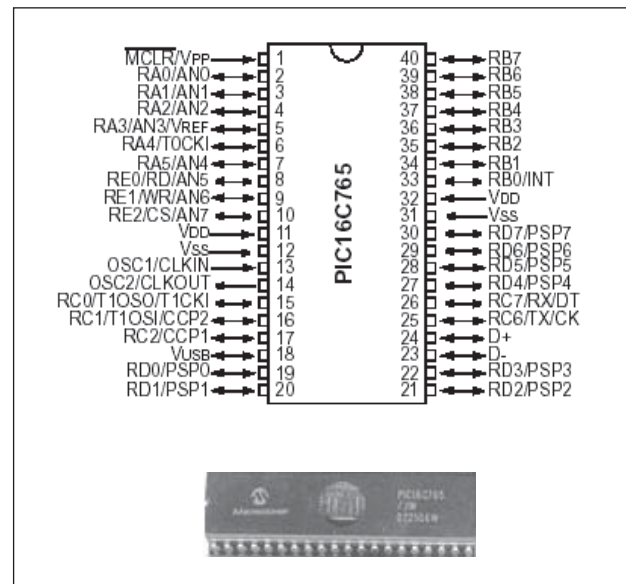


Figura 9. Microcontrolador PIC16C765

Tabla 4. Especificaciones del microcontrolador USB

	PIC16C745	PIC16C765
Frecuencia de operación	6 MHz ó 24 Mhz	6 MHz ó 24 MHz
Resets (y retardos)	POR, BOR (PWRT, OST)	POR, BOR (pwrt, ost)
Memoria de programa	8 k	8 k
Memoria de datos	256	256
Memoria RAM dual (puerto)	64	64
Fuentes de interrupción	11	12
Puertos I/O	22 (puertos a, b, c)	33 (puertos a, b, c, d, e)
Temporizadores	3	3
Captura/ comparador /módulo PWM	2	2
Módulo de conversión análogo a digital	5 canales x 8 bit	8 canales x 8 bit
Puerto esclavo paralelo	No	Sí
Comunicación serial	USB, USART/SCI	USB, USART/SCI
Brown – out detect reset	Sí	Sí

4.4 Elaboración del prototipo final

En la figura 10 se encuentra una foto de la tarjeta final, diseñada bajo el software para la elaboración de circuitos impresos Tagle. El circuito impreso es doble faz con hueco pasante metalizado, logrando así un tamaño relativamente pequeño.

Los PIC16C745 y 16C765 cuentan con una salida de voltaje de 3,3 V VUSB, en la cual se conecta una resistencia de *pull-up* y un condensador, para contar con los canales adecuados de comunicación con un *host* (véase figura 11).

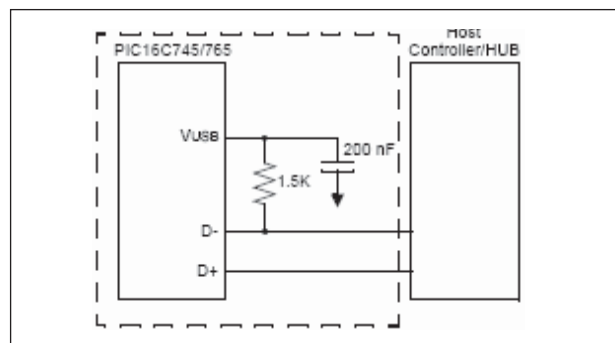


Figura 11. Microcontrolador al *host* USB

El prototipo sólo consume 100 mA a 5 V, razón por la cual se pudo alimentar del mismo cable USB⁴. En la figura 12 se muestra el diagrama de flujo correspondiente al programa fuente del microcontrolador; en él se visualizan las etapas de inicialización de comunicaciones, conversión de señal y espera de transmisión de datos mediante USB.

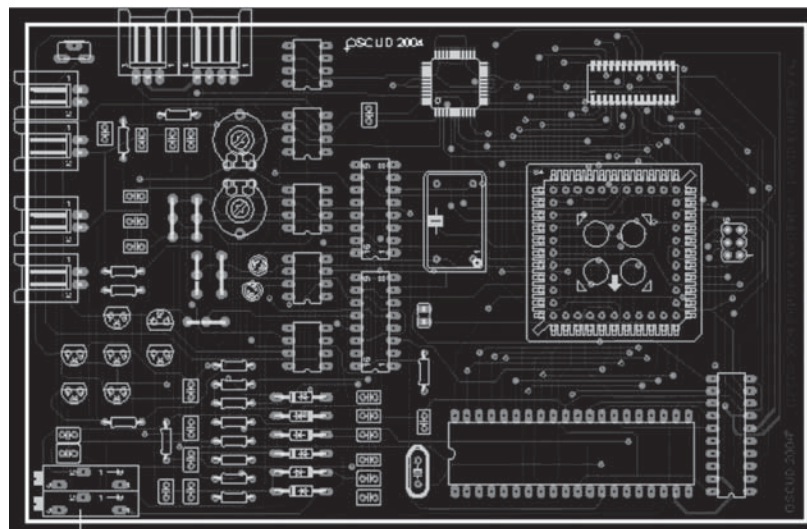


Figura 10. Tarjeta de circuito impreso

4.4.1 Software de adquisición de datos

(véanse figuras 13 y 14).

Fue diseñado en Visual Basic y se utilizó el control ActiveX suministrado por Microchip, el cual identifica y controla la comunicación con sus microcontroladores. Este sistema de control no es la única solución para la comunicación con los dispositivos USB; también existe la opción de trabajar con el API de Windows, sobre el cual pueden diseñarse controladores DLL y adicionarlos a una aplicación de software específica.

⁴ Voltaje proveniente del *host*.

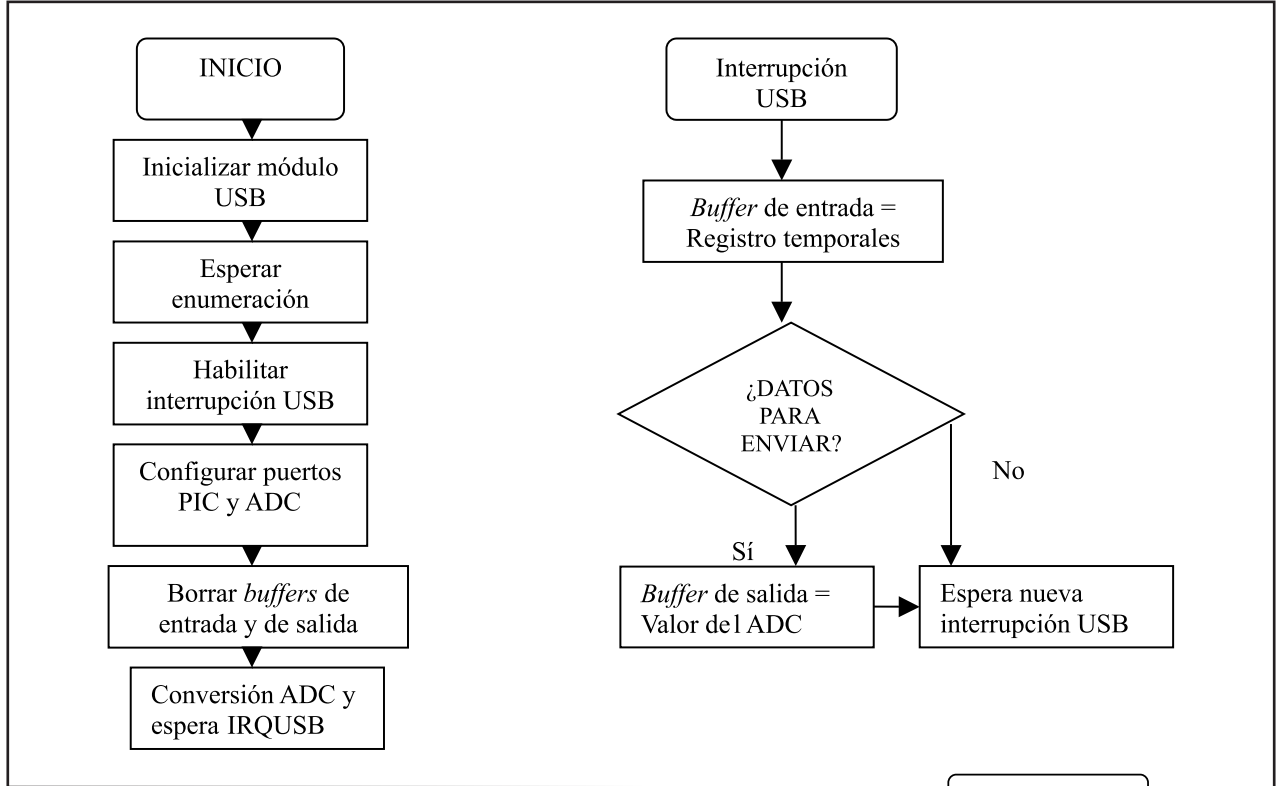


Figura 12. Diagrama de flujo del *firmware*

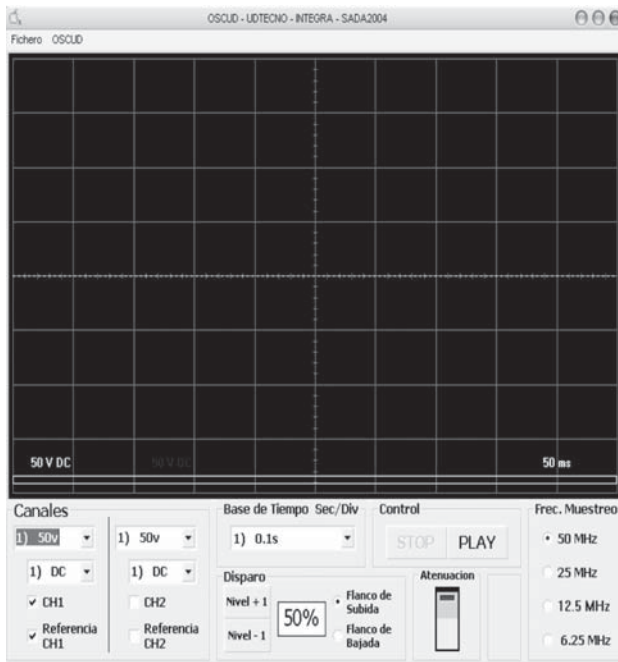
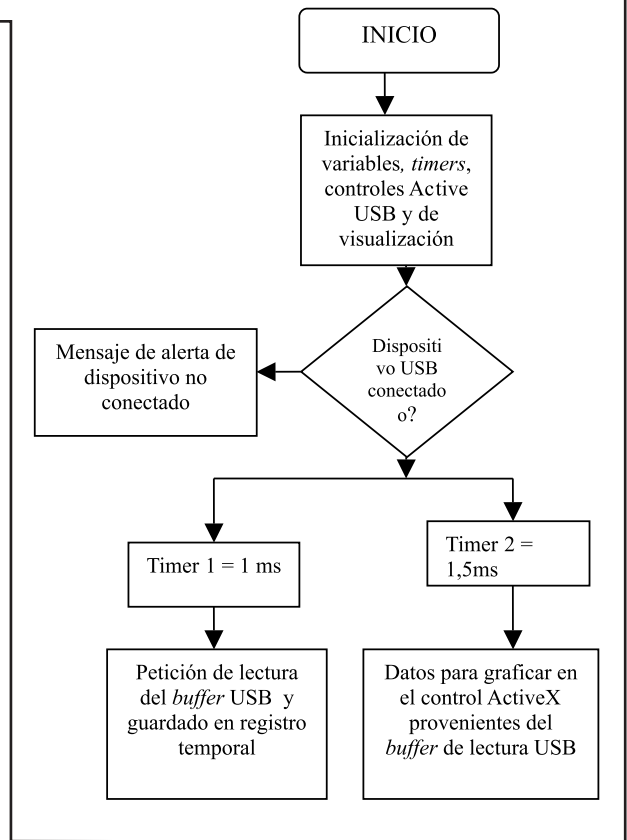


Figura 13. Software USB Osciloscopio UD (OSCUD)



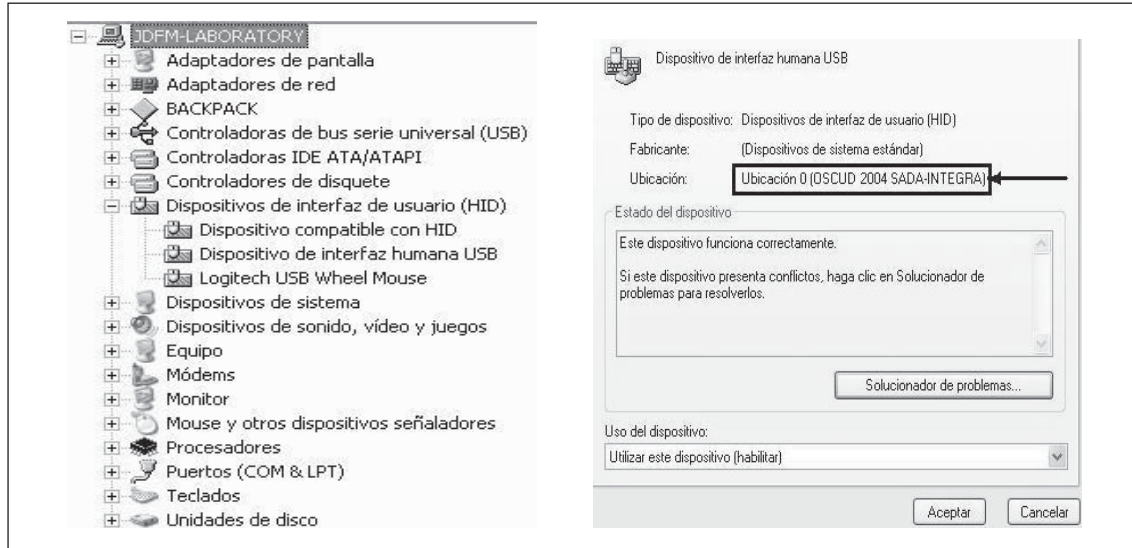


Figura 14. Capturas de pantalla cuando se identifica el dispositivo USB sobre Windows Xp

El ActiveX sólo funciona para dispositivos USB del tipo HID. Los dispositivos HID no necesitan controladores específicos, dado que Windows o el sistema operativo los identifican y enumeran automáticamente (*mouse*, teclado, mandos de juego, entre otros).

4.5 Método de muestreo

Los osciloscopios digitales pueden reunir más puntos de los necesarios para reconstruir posteriormente la señal en la pantalla. No obstante, para señales rápidas (alta frecuencia) el osciloscopio no puede recoger muestras suficientes y debe emplear una de estas técnicas:

- Interpolación: estimar un punto intermedio de la señal basándose en el punto anterior y posterior.
- Muestreo en tiempo equivalente: si la señal es repetitiva, es posible muestrear durante unos cuantos ciclos en diferentes partes de la señal para después reconstruir la señal completa. Este tipo de muestreo fue el implementado en el diseño y la elaboración del osciloscopio.

4.5.1 Muestreo en tiempo equivalente

Algunos osciloscopios digitales lo utilizan. Se trata de reconstruir una señal repetitiva (periódica) capturando una pequeña parte de la señal en cada ciclo. Existen dos tipos básicos: a) muestreo secuencial: los puntos aparecen de izquierda a derecha en secuencia para conformar la señal; b) muestreo aleatorio: los puntos aparecen aleatoriamente para formar la señal.

4.6 Resultados obtenidos

- Comunicación del microcontrolador con el computador por medio del bus (puerto) USB, garantizando todas sus especificaciones (identificación, enumeración, etc.) Véase figura 14.

Figura 15 (a y b). Capturas de pantalla en mediciones de prueba del proyecto “Osciloscopio UD”

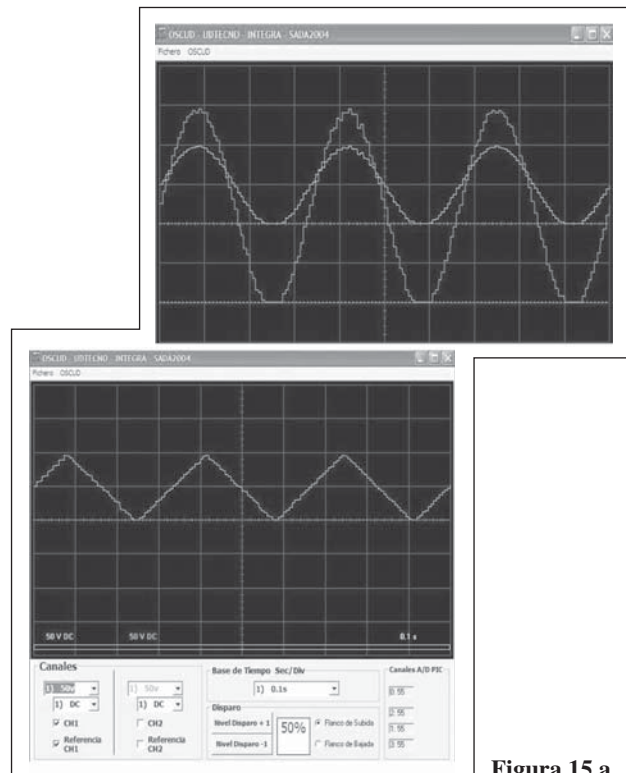
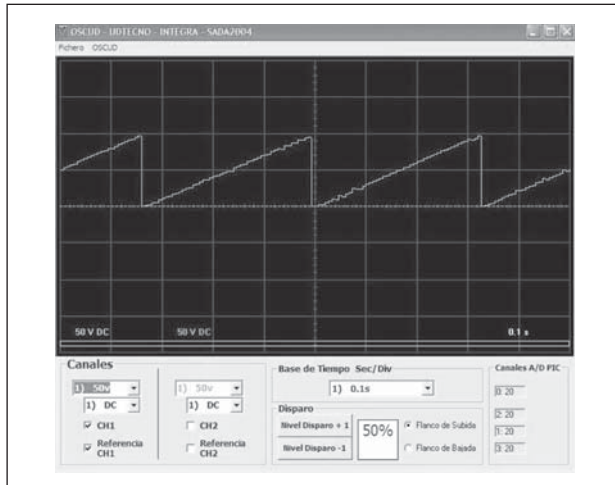


Figura 15 a

Figura 15 b



Las figuras 15a y 15b muestran mediciones realizadas durante las fases de prueba del proyecto; ellas corresponden a señales de frecuencia fija y diferente forma de onda.

- Se logró una comunicación estable a una velocidad promedio de 800 kbps, realizando pruebas en sistemas operativos Windows 98 SE, Windows Millenium, Windows 2000, Windows XP Professional y Home, y versión Beta de Windows Long Horn.

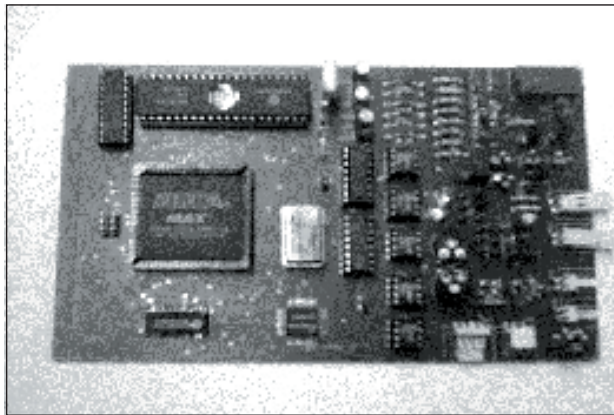


Figura 16. Prototipo obtenido

En la figura 16 se muestra el circuito implementado para el desarrollo de las pruebas y como prototipo final. En este momento se realizan pruebas tendientes a mejorar el ancho de banda y otras características propuestas en el anteproyecto.

5. Conclusiones

- Con el desarrollo de este proyecto se busca fortalecer en el medio colombiano el diseño y uso de

la instrumentación virtual como una alternativa de bajo costo, acorde con las necesidades técnicas del sector industrial y académico en cuanto a empleo de instrumentos de medida.

- Como primer acercamiento al empleo de la tecnología de comunicación USB, el microcontrolador utilizado respondió a las expectativas; se afianza de esta manera el conocimiento de sus potencialidades.
- Para futuras aplicaciones que necesiten mayor velocidad de transmisión; se recomienda utilizar dispositivos de alta velocidad (Cypress de alta velocidad).
- La universidad colombiana puede contribuir al desarrollo y la apropiación de las nuevas tecnologías en instrumentación y de las tecnologías avanzadas de diseño digital.
- El osciloscopio UD se convierte en la primera aplicación desarrollada en la Universidad Distrital Francisco José de Caldas de Bogotá D.C., y en una de las primeras aplicaciones que a nivel nacional utiliza el bus de comunicaciones USB para instrumentación virtual.
- Se espera que este desarrollo se convierta en herramienta para la divulgación del uso de la instrumentación virtual en la industria y la academia, facilitando la comprensión de una interfaz de comunicaciones tan importante en la actualidad, cual es el bus USB.

Referencias bibliográficas

- [1] MING TANG, W. (1997). *Developing USB PC peripherals*, AnnaBooks.
- [2] ANDERSON, D. (2001). *USB System Architecture (USB 2.0)*, Boston, Addison-Wesley Developers.
- [3] AXELSON, J. (2001). *USB Complete*. 2nd ed., Madison, Press Lakeview Research
- [4] HYDE, J. (2001). *USB Design by Example*. Independent Publisher Group.
- [5] FONSECA, A., FINO, R. (2003). *Red de datos para instrumentación*, Bogotá, Fondo de Publicaciones Universidad Distrital Francisco José de Caldas.
- [6] LÁZARO, A. M. (2001). *LabView 6i Programación grafica para el control de instrumentación*, Madrid, Ed. Thomson Paraninfo.
- [7] PREDKO, M. (2000). *Programming and Customizing PICmicro Microcontrollers*, 2nd ed., Ed. McGraw-Hill.
- [8] Compaq, Digital Equipment Corporation, IBM PC Company, Intel, Microsoft, NEC, and Northern Telecom, "USB Specification 1.0", Revision 1.0, Jan 19, 1996. <http://www.usb.org>
- [9] USB IMPLEMENTERS FORUM JF2-51, 2111 NE 25th Avenue Hillsboro, OR 97124, tomado del world wide web: <http://www.usb.org>
- [10] "8x930Ax, 8x930Hx USB Microcontroller User's Manual", Corporation Literature Sales P.O. Box 7641 Mt. Prospect, IL 60056-7641. <http://www.intel.com/design/usb/manuals>
- [11] "8x93QAx USB Microcontroller Datasheet", order no: 272917-003, February 1997, and the "8x930Ax Specification Update", order no: 272940 available at <http://www.intel.com/design/usb>