

Adquisición, registro y transmisión en tiempo real de señales sismológicas bajo TCP/IP

CARLOS A. VARGAS-JIMÉNEZ¹, G. CASTELLANOS², J. A. FLÓREZ², J. P. GÓMEZ²

RESUMEN

Se presenta el desarrollo de un sistema de monitoreo de variables sísmicas, que incluye la adquisición y adecuación de 16 canales simultáneos, el preproceso, la grabación y la transmisión sobre redes IP en tiempo real. El sistema está orientado al sistema Windows, para lo cual se emplearon librerías dinámicas orientadas al acceso de dispositivos en tiempo real. Así mismo, el acceso de datos al PC se realiza vía DMA, para implementar el proceso múltiple de señales en tiempo real. La arquitectura de la red IP de monitoreo sísmico es cliente/servidor, en la cual se tomó como servidor un PC dotado de una tarjeta de adquisición de datos, encargado de realizar la conversión análogo-digital de las señales, almacenar en archivos las que correspondan a eventos sísmicos y, al mismo tiempo, atender las solicitudes de los diferentes clientes. El software de cliente permite a los usuarios visualizar en tiempo real las señales que el servidor está adquiriendo y realizar un procesamiento básico de las señales registradas en el servidor.

PALABRAS CLAVE: ADQUISICIÓN DE DATOS, TCP/IP, TIEMPO REAL

ABSTRACT

The constant evolution of network technologies has made possible the development of applications facilitating the real time access to the information. This way, it is possible the implementation of remote monitoring systems that using PCs achieves efficient transmission of signals through computer networks. In this work, the procedures system design and implementation procedures are explain in Client/Server environment that, based on TCP/IP, transmits in real time 16 seismological signals over LAN network and the Internet. The server is a PC endowed with an acquisition data card and it is the one in charge of carrying out the signal Analog/Digital conversion, to store in files those that correspond to seismic events and, at the same time, to assist the applications of the different clients. The client's software allows the users to view in real time the signals that the server is acquiring and to carry out a basic processing to the signals that have been registered in the server.

KEY WORDS: DATA ACQUISITION, TCP/IP, REAL TIME

INTRODUCCIÓN

El auge de los PC, y especialmente de Internet, en los últimos años ha permitido el desarrollo de sistemas que monitorean en tiempo real señales que representan variables físicas a un bajo costo. En este caso se trata de señales sísmicas capturadas por un geófono, el cual genera un voltaje de acuerdo con la velocidad del movimiento del suelo. Estos sensores se encuentran en diferentes ubicaciones geográficas, por lo que se hace necesario utilizar enlaces telemétricos para centralizar la información. Una vez agrupadas las señales, en el servidor se efectúa la conversión análogo-digital. En este artículo se desarrolla un sistema de monitoreo sísmico que contempla la adquisición, el preproceso, el registro de señales sísmicas y su posterior envío y acceso a través de redes de IP. La transmisión de las señales se realiza bajo el protocolo TCP/IP, logrando que el software opere en Intranets e Internet sin necesidad de modificaciones en su estructura. Así mismo se detalla el funcionamiento de los algoritmos de procesamiento digital de señales implementados en el software de cliente para dar herramientas a los sismólogos que estudian las señales adquiridas remotamente en el servidor.

ADQUISICIÓN DE SEÑALES SISMOLÓGICAS

Las señales sismológicas son consideradas como no estacionarias, y su ancho de banda está entre 0.001 y 50 Hz, por lo que una frecuencia de conversión de 100 muestras por segundo por canal es considerada suficiente, restricción fácilmente superable para el desarrollo de sistemas de proceso en tiempo real, incluida su transmisión. En general, hay tres formas de transferir los datos del dispositivo de adquisición a la memoria en la arquitectura del PC (Hazzah, 1997): por consultas o *polling*, por interrupciones y por acceso directo a memoria (DMA, *Direct Memory Access*). De estas tres técnicas, la adquisición por DMA es la que ofrece mayores ventajas para aplicaciones que deben garantizar el muestreo de una o varias señales a una frecuencia constante. El DMA permite que dispositivos externos transfieran información directamente desde o hacia la memoria del sistema, sin la intervención del procesador, mediante su respectivo controlador, el cual se apodera temporalmente del manejo de los buses (de datos, direcciones y control) del microprocesador, y transfiere los datos directamente del puerto o de la memoria, a localidades de memoria específicas, o viceversa.

El DMA tiene como propósito principal liberar a la unidad central de procesamiento (CPU) de la carga de atender a algunos controladores de dispositivos, opción que no se tiene en los casos de adquisición por consultas o por interrupciones. El DMA actúa como una unidad de procesamiento secundaria, toma el control de los buses de datos, direcciones y

¹ Universidad Nacional de Colombia, Departamento de Geociencias, Grupo de Geofísica, Bogotá.

² Universidad Nacional de Colombia, Grupo Control & Procesamiento Digital de Señales.

control, se encarga de capturar el número de muestras con que fue programado y los almacenan a partir de la posición especificada (García de Celis, 1999). De esta manera, DMA es un método que proporciona el multiproceso por hardware.

La tarjeta de adquisición de datos que se conecta al PC que desempeña el papel de servidor debe cumplir los siguientes requerimientos: conversión simultánea de 16 canales, resolución de 12 bits, frecuencia de conversión superior a 100 mps y posibilidad de realizar transferencias DMA. En el mercado se encuentran disponibles diferentes modelos de tarjetas que cumplen con estos requerimientos. En el caso particular, el software de servidor fue desarrollado utilizando una tarjeta National Instruments modelo PCI MIO 16E4 (National Instruments, 1997).

Sin embargo, el manejo de dispositivos desde el sistema Windows a 32 bits no es directo, y es necesario el empleo de DLL (*Dinamic Linked Library*) especializadas. En particular se emplea el VxD (*Virtual Device Driver*), que tiene acceso irrestricto a todos los dispositivos de hardware, examina libremente las estructuras de datos del sistema operativo y puede acceder cualquier dirección de memoria (Hazzah, 1997). Así, la VxD puede manejar interrupciones y realizar transferencias por DMA. Para la tarjeta escogida, se proporcionan las VxD en forma de API (*Application Programming Interface*), que se compone de una librería de rutinas que permite programar la tarjeta y configurar parámetros como la frecuencia de conversión, el rango de las señales de entrada y el manejo del buffer, entre otros.

El lenguaje de programación utilizado fue el LabVIEW 6i, el cual es un lenguaje de programación gráfico orientado a la instrumentación (National Instruments, 2000). Desde la aplicación hecha en LabVIEW se accesa el hardware a través de la API mencionada. Las funciones contenidas en las DLL de la API permiten realizar transferencias DMA. El método como se transfieren los datos de la tarjeta de adquisición a la memoria es definitivo para poder garantizar una frecuencia de conversión constante.

Por ser Windows a 32 bits un sistema operativo de multitarea apropiativa, no se puede garantizar la ejecución de una rutina de tiempo real crítico (*hard real-time*), como es en este caso la conversión A/D a una frecuencia constante. Este requerimiento se logra realizando la tarea de tiempo real crítico por hardware, es decir que, una vez programada la tarjeta de adquisición, ésta realiza autónomamente las conversiones A/D, transfiriendo los datos a un buffer DMA. Debido a que la tarjeta es para bus PCI, se puede manejar Bus-master DMA, que no tiene tantas restricciones como las transferencias DMA para bus ISA (Hazzah, 1997). Con Bus-master DMA se pueden manejar apuntadores a direcciones por encima de las 16Mb, y además el buffer no tiene que ser consecutivo (*scatter-gather DMA*). Una vez cumplidos los requerimientos de tiempo real, la aplicación que se ejecuta en el servidor toma las muestras del buffer, que han sido puestas en éste, en virtud de la transferencia *Bus-master DMA*.

Para el caso de las señales sísmicas es necesario contar con información de un cierto número de muestras anteriores a la declaración del evento. Esto se debe a que los algoritmos que declaran el evento, activan el comando de almacenamiento varias muestras después del inicio del sismo, cuando se ha superado un umbral de sensibilidad elegido por el usuario. Se recomienda un pre-evento de alrededor de 30 s, que puede variar dependiendo del tipo de estudios que se realicen en la red sísmica. La duración exacta del pre-evento es especificada por el usuario una vez se inicia el proceso de conversión A/D; de esta forma, si se considera que se muestrea la señal 100 veces por segundo y se define un pre-evento de 30 segundos, entonces el buffer tiene una longitud de 3000 palabras por canal, ya que se necesitan dos bytes (una palabra), para almacenar cada muestra convertida a una resolución de 12 bits. La figura 1 muestra la señal digitalizada en el servidor, expresada en cuentas. Para una resolución de 12 bits y señal bipolar, se tiene un rango que va desde -2048 a 2048 cuentas. Una cuenta equivale a 2.44 mV, considerando un rango para la señal analoga de -5 a 5 voltios.

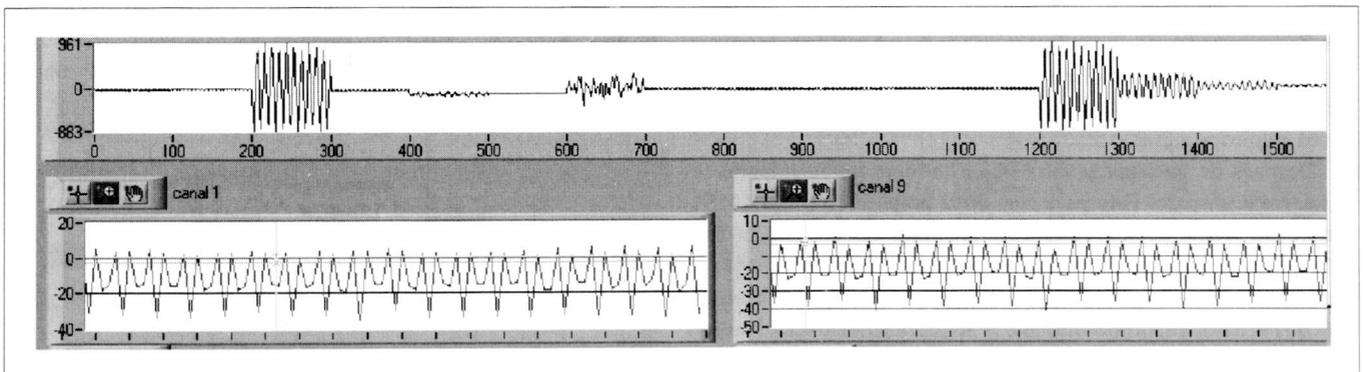


Figura 1. Ejemplo de señales adquiridas en el servidor

REGISTRO DE EVENTOS SÍSMICOS EN EL SERVIDOR

Una red sísmica o incluso una sola estación sísmica que opera continuamente a ciertas frecuencias y con una alta resolución, es este caso 12 bits, produce una cantidad enorme de datos que son a menudo difíciles de guardar y analizar. El objetivo de los algoritmos de disparo es detectar las señales sísmicas típicas (terremotos, fuentes controladas

de señales sísmicas, explosiones nucleares, señales subterráneas, etc.), en las cuales aparece ruido sísmico constantemente. Una vez se detecta un supuesto evento sísmico, el algoritmo guarda todas las señales entrantes y sólo se detiene después de que se declara el fin del evento, o después de un tiempo predeterminado. Los algoritmos automáticos de disparo son relativamente ineficaces cuando se comparan

con la experiencia humana, así, éstos pueden generar archivos falsamente activados por supuestas señales sísmicas o, por el contrario, no almacenarán pequeñas señales detectadas como ruido sísmico; por tales motivos la sensibilidad del filtro estará dada por las necesidades de la red sísmica. En el servidor se utiliza el algoritmo de disparo STA/LTA (*Short Term Average/Long Term Average*), que realiza procesos de filtración de las señales sísmicas. El STA mide la amplitud del instante de la señal sísmica, mientras el LTA cuida del promedio actual de la amplitud del ruido sísmico. La ecuación en diferencia que permite describe el STA/LTA es (Lee, 1989):

$$y(n) = \frac{1}{k} x(n) + \frac{k-1}{k} \cdot y(n-1) \quad (1)$$

donde x representa la entrada, y el promedio computado, k es el tamaño del STA o del LTA, n representa el período de tiempo actual y $n-1$ el período de tiempo previo. La ecuación en diferencia (1) implementa un filtro pasabajos normalizado, donde la frecuencia de corte es controlada por k . Un k grande representa una pequeña frecuencia de corte y un gran tiempo de promedio. Para determinar k si se ha especificado la frecuencia de corte, se debe resolver la siguiente ecuación (Lee, 1989):

$$a^2 + (2 \cdot \cos(w \cdot T) - 4)a + 1 = 0$$

donde $a = 1/k$, w es la frecuencia de corte en radianes y T es el período de muestreo. Para aplicar el STA/LTA, se calcula la amplitud absoluta de cada dato de la señal entrante. Luego, se calcula el promedio de amplitudes absolutas en ambas ventanas. En el próximo paso se calcula una proporción de ambos valores de STA/LTA; esta proporción se compara continuamente a un valor seleccionado de umbral; si la proporción excede este umbral, se declara un evento. Esto no necesariamente significa que la red empieza a grabar señales sísmicas, puesto que realmente en el sistema se tienen varias estaciones. Todas las redes sísmicas y la mayoría de los registradores sísmicos tienen un mecanismo de disparo que define cuántos y qué canales tienen que estar en un estado activado antes de que el instrumento o la red empiecen a grabar datos.

Después de que una señal sísmica termina gradualmente, se desactiva el disparo. Esto pasa cuando la proporción de STA/LTA actual cae debajo de otro parámetro que entra el usuario. El STA/LTA de desactivación debe tener un nivel de umbral más bajo (o raramente igual) que los STA/LTA que activan el nivel de umbral. Además de los datos adquiridos durante la activación del disparo, las redes sísmicas y los registradores sísmicos agregan una cierta cantidad de datos sísmicos al evento archivado que corresponden al pre-evento de tiempo (PEM), y después de que el estado activo termina, el sistema también agrega un post-evento de tiempo (PET).

En la figura 2 se presenta una señal sísmica y los cálculos de STA y de LTA. En la segunda gráfica se observa un crecimiento lento pero constante del LTA y una variación del STA que sigue de una forma rápida los cambios en la señal; de esta forma se declara el evento cuando la razón STA/LTA supera el umbral predefinido.

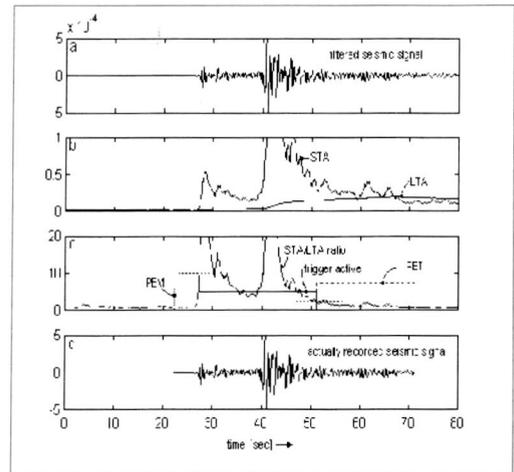


Figura 2. Ejemplos de promedios STA/LTA y activación de disparos

TRANSMISIÓN EN UN ENTORNO DE RED EN TIEMPO REAL

El auge de Internet y las intranets ha convertido a TCP/IP en el protocolo de comunicaciones más ampliamente usado en las redes de computadores a nivel mundial. Por esta razón todos los sistemas que utilicen redes de computadores como medio de transmisión están tendiendo a utilizar TCP/IP como protocolo de comunicación (Parker, 1997). Los sistemas de instrumentación no son la excepción. De esta forma, los usuarios pueden tomar datos al tiempo que son adquiridos en el servidor, accediendo la información en tiempo real, sujeto a la velocidad de la red. En el sistema desarrollado para el presente trabajo, el servidor debe estar en capacidad de transmitir 16 señales digitalizadas con una resolución de 12 bits y a 100 muestras por segundo, lo que determina que se debe estar transmitiendo como mínimo 25.6Kbps. Esta velocidad de transmisión determina la velocidad mínima necesaria, pero no considera los bits que agrega el protocolo de transmisión de datos. Otra consideración de diseño es que el servidor debe estar en capacidad de atender peticiones simultáneas de diferentes clientes conectados a la red.

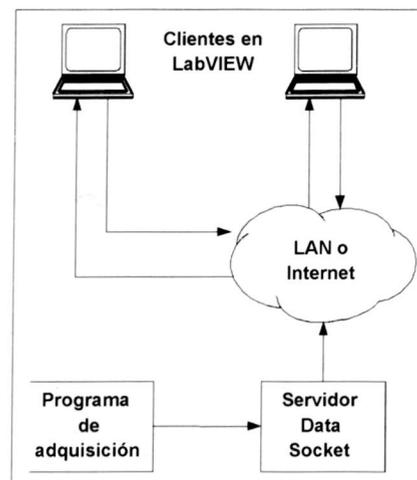


Figura 3. Modelo cliente/servidor

En la figura 3 se ilustra el esquema de conexión entre el proceso de adquisición en el servidor y los clientes. Esta conexión se basa en la utilización de *sockets*, los cuales son puntos finales de enlaces de comunicaciones entre procesos. Es decir que cada circuito que entre y sale de la capa TCP se identifica de manera única por una combinación de números, o *sockets*. Estos números son la dirección IP de la máquina y el número de puerto usado por el software TCP, haciendo posible que un proceso hable con otro proceso a través de la red, basado por completo en el número de *socket*. La tecnología Datasockets provee al programador una API para transmisión de señales a través de una red, utilizando URL (*Uniform Resource Locator*). Cada URL identifica explícitamente un único proceso en una máquina específica, por ejemplo una URL puede ser `dstp://200.30.41.75/datos`.

El modelo de red cliente/servidor está constituido por aplicaciones cliente que requieren los servicios que proporciona la máquina servidor. La aplicación de servidor se dedica a escuchar las direcciones conocidas (puertos de servicio) para el requerimiento de servicios. Cuando una petición tiene lugar, el proceso servidor despierta de su estado latente y atiende la dirección solicitada ejecutando las acciones específicas que sean procedentes. En esta aplicación se trabaja con 16 señales, por lo que se decidió construir una trama propia con estas señales y agregarle un corto encabezado, que contiene el número de canales enviados y otra in-

formación necesaria para que el cliente interprete cada señal sísmica. Esta única trama se escribe a un *socket*, que previamente ha sido abierto.

RESULTADOS Y CONCLUSIONES

El resultado final es una aplicación en ambiente Windows que adquiere hasta 16 señales sísmicas, las procesa y registra acorde a los resultados del algoritmo de STA/LTA.

El sistema de adquisición fue probado satisfactoriamente con las señales que entrega la red sísmica regional eje cafetero, en las instalaciones de Ingeominas-Manizales, en Colombia. Esta red sísmica esta dotada de registradores electromecánicos que generan impresiones continuas de todas las estaciones. De esta manera se puede efectuar una comparación visual entre los archivos generados por el sistema de adquisición digital y los registros el papel. Se observó que los sismos emergentes que crecen lentamente no son siempre detectados por el algoritmo STA/LTA. Por esa razón fue implementado un algoritmo de detección más simple, basado en un umbral de amplitud. Así, el usuario puede elegir la declaración de evento por STA/LTA o por umbral.

La figura 4 muestra las señales recibidas en un cliente conectado al servidor a través de una red LAN a 10Mbps. Cinco canales indican señales de ruido sísmico y uno de los canales registra una sinusoidal de prueba entregada por un generador de señales.

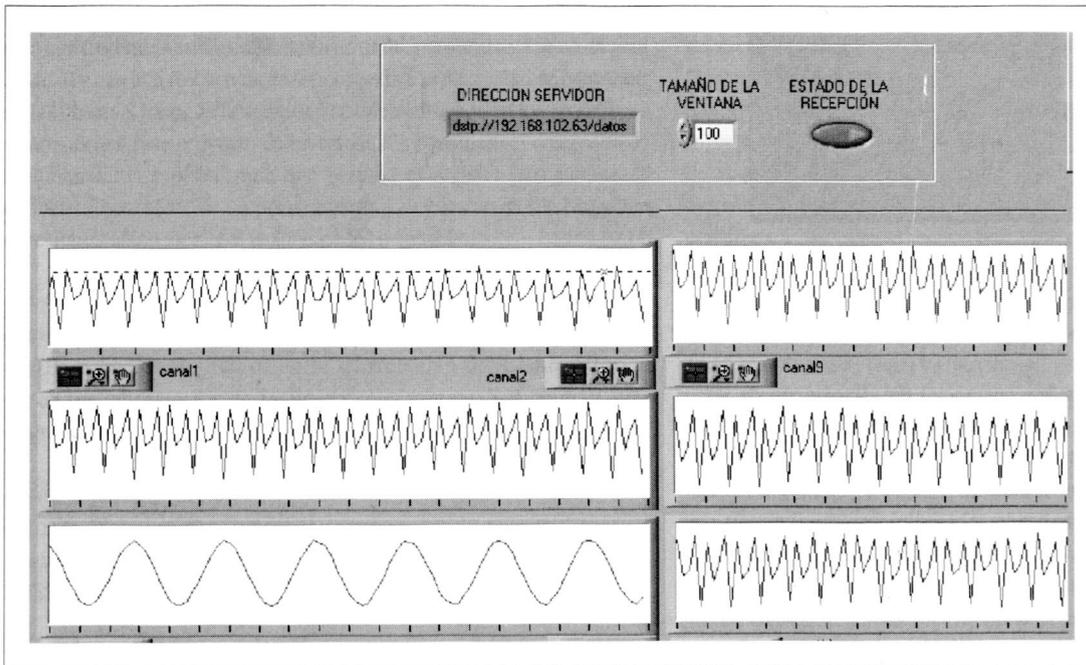


Figura 4. Recepción de señales en el cliente

Para el usuario, el proceso de conexión es totalmente transparente. Una vez se cuente con la infraestructura de red, sólo es necesario conocer la dirección IP del servidor y el nombre del proceso.

Las herramientas de software implementadas que dispone el cliente se agrupan en tres categorías: preprocesamiento en el dominio del

tiempo, análisis en el dominio de la frecuencia y aplicación de filtros. El programa permite aplicar las siguientes funciones de ventana a sismos que se cargan desde archivos: Hanning, triangular, Blackman, Hamming, Blackman exacta, coseno tapered y Blackman-Harris. Así mismo puede utilizar filtros Chebyshev, Butterworth y elípticos. Es-

tos filtros permiten variar sus frecuencia de corte, la clase: pasabajos, pasaaaltos, pasabanda o rechazabanda y el orden del mismo. En el dominio de la frecuencia se pueden realizar FFT, auto-espectro de potencia, el espectro cruzado de potencia y la transformada de Hilbert. La figura 5 detalla las herramientas de procesamiento digital de señales que posee el software de cliente.

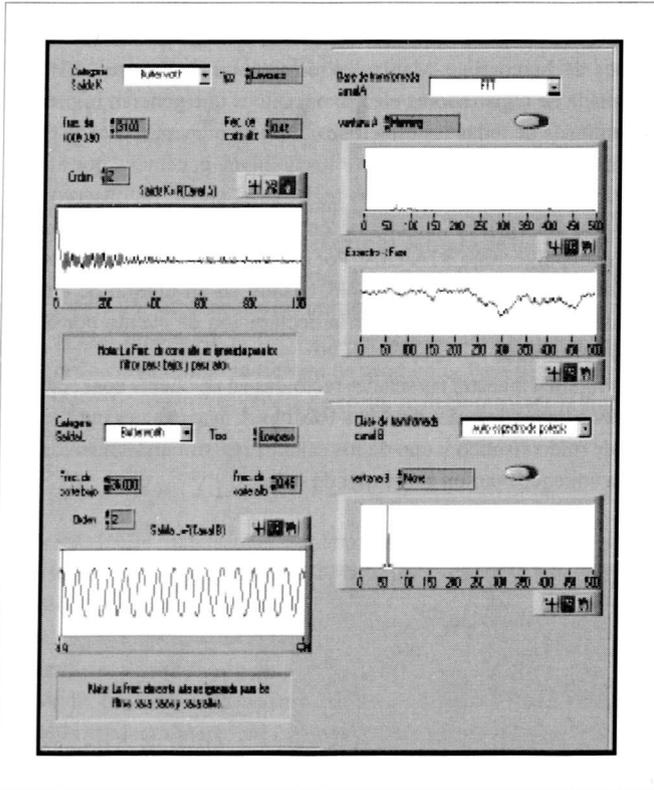


Figura 5. Procesamiento digital de señales efectuado por el software de cliente

Se observó también una mejora sustancial del desempeño de las rutinas de procesamiento en tiempo real, cuando se trabaja con números expresados en formato entero, ya que una operación entre números expresados en coma flotante consume más tiempo de ejecución de la CPU que la misma operación realizada sobre números enteros.

De igual forma se anota que almacenar los datos adquiridos en formato binario presenta más ventajas que el almacenamiento en ASCII. Almacenar en formato binario es más rápido que en ASCII: esta ventaja es importante debido a la característica de tiempo real de la aplicación. Por otra parte, los archivos binarios son de menor tamaño y no se pierde precisión al realizar la conversión a ASCII.

REFERENCIAS BIBLIOGRÁFICAS

- AKI, K., RICHARDS, P. 1980: Quantitative Seismology Theory and Methods, W.H. Freeman and Company.
- CADZOW, J. A. 1973: Discrete Time Systems, Prentice Hall, Englewood Cliffs, N.Y.
- F. HARVEY. 1994: DMA Fundamentals on Various PC Platforms, National Instruments.
- GARCÍA de Celis, C. 1999: El universo digital del IBM PC, cuarta edición.
- HAZZAH, K. 1997: Writing Windows VxDs and Device Drivers, R&D Books.
- NATIONAL INSTRUMENTS. 1997: PCI-MIO E Series User Manual Multifunction I/O Boards for PCI Bus Computers.
- LEE, W. H. K. 1989: ToolBox for Seismic Data Acquisition, Processing and Analysis, International Association of Seismology and Physics of the Earth's Interior.
- NATIONAL INSTRUMENTS. 2000: LabVIEW User Manual National Instruments, LabVIEW version 6i-Internet-Ready Measurement Applications.
- PARKER, T. 1997: Teach yourself TCP/IP in 14 days, segunda edición, Prentice Hall.
- PROAKIS, J., MANOLAKIS, D. 1998: Tratamiento digital de señales, tercera edición, Prentice Hall.
- TAYLOR, F. Mellott, J. 1998: Hands-on Digital Signal Processing, McGraw-Hill.
- TMKOCZY, A. 1999: Understanding and Setting the STA/LTA Trigger Algorithm and Associated Parameters, <http://www.seismo.com/msop>