



Universidad Tecnológica de la Mixteca

ADS

Analizador Dinámico de Señales

Tesis Profesional

Que para obtener el Grado de

Ingeniero en Electrónica

Presenta

Heriberto Ildefonso Hernández Martínez

Directores de tesis:

Ing. Hugo Suárez Onofre

Fis. Mat. Gustavo Jiménez Santana

Acatlima, Huajuapán de León, Oaxaca

Julio'96

**Tesis presentada el 25 de Julio de 1996
ante los siguientes sinodales:**

**M.C. José A. Moreno Espinosa
M.C. Albino Martínez Sibaja
Ing. Misael Huerta López**

Asesores:

**Ing. Hugo Suárez Onofre
Fis. Mat. Gustavo Jiménez Santana**

■ Dedicatorias ■

A mi madre, por toda la fuerza y amor en estos años juntos.

A mi padre, por todo ese silencio que me ha comunicado todo.

A *Refugio*, por darnos la oportunidad de pensar en él.

A *Francisco*, por su ejemplo y valor para librar obstáculos.

A *Guadalupe y Vianey*, por llenar de cariño el espacio que faltaba.

Heriberto.

Agradezco especialmente a:

Ing. Hugo Suárez Onofre

Fis. Mat. Gustavo Jiménez Santana.

Por su confianza y ayuda para
realizar este trabajo.

Heriberto I. Hernández Martínez

■ Agradecimientos ■

A todos mis familiares, mis abuelos, tíos y primos, lejos pero cerca de mi corazón.

A todos mis profesores, por compartirme sus conocimientos.

Al M.C. José A. Moreno Espinosa, por brindarme un poco de su tiempo para realizar mejoras a mi vida universitaria.

A mis compañeros de generación -90-95-, en especial a Héctor D. Alfaro Pérez, por viajar en el mismo barco juntos.

A las siguientes personas, cuya ayuda fue un alisiente para lograr este objetivo: Ing. Gerardo García hernández, Pbro. José Gonzalez Acevedo, Prof. Leonardo Ramírez Rivera, Sra. Carmela Martínez Vargas, Ing. Sabino A. Hernández Durán, Prof. Rogelio Rivera García, Lic. Gabriel Rivera García, Ing. Alejandro Ruiz Figueroa, Ing. Guillermo Calero Ortiz, Profa. VedaBertha Córtez Rios, Profa. Mixteca Lara Ortiz, Fam. Gómez Corzo, Fam. Hernández Martínez, Héctor Espinosa Ramírez, Alfredo Higuera Cortés, Ing. Conrado García Olivares.

Heriberto I. Hernández Martínez

Resumen

Este documento presenta las fases de desarrollo del proyecto de tesis Analizador Dinámico de Señales. Desde el análisis de la necesidad de este tipo de sistemas hasta la documentación que respalda se aplicación, se incluyen las etapas de conceptos generales, diseño e implementación del sistema ADS. Así como las conclusiones, alcances y tendencias.

El proyecto de tesis del Analizador Dinámico de Señales se planteó como un sistema con componentes hardware y software para elaborar un instrumento que simule el funcionamiento de un Osciloscopio y el de un Analizador de Espectros; en base al diseño de una tarjeta de adquisición de datos y un conjunto de librerías elaboradas en lenguaje C++.

Indice

Dedicatorias	i
Agradecimientos	ii
Resumen	iv
Introducción	viii
1. Consideraciones generales	1
1.1. Procesamiento digital de señales	3
1.1.1. Procesamiento digital de señales en instrumentación	5
2.1. Instrumentación virtual	9
1.2.1. Fases de diseño de un instrumento virtual	11
1.2.2. Elementos de un instrumento virtual	14
2. Análisis matemático de señales	19
2.1. Teorema de muestreo	20
2.2. Análisis de Fourier	25
2.2.1. Series de Fourier	26
2.2.2. Transformada de Fourier	29
2.2.3. Transformada discreta de Fourier (DFT)	30
2.2.4. Transformada rápida de Fourier (FFT)	33
2.3. Conversión analógica a digital (A/D)	40
2.3.1. Relación Entrada / Salida	41
2.3.2. Errores de conversión	42
2.3.3. Resolución del convertidor	43
2.3.4. Precisión del convertidor	43
2.3.5. Tiempo y velocidad de conversión	43
2.4. Convertidor A/D por aproximaciones sucesivas	46
2.4.1. Operación del circuito	46
2.4.2. Analogía por aproximaciones sucesivas	47
2.4.3. Tiempo de conversión por aprox. Sucesivas	48
3. Adquisición y presentación de datos	49
3.1. Acondicionamiento de la señal	51
3.2. Diseño de la tarjeta de adquisición de datos	53
3.2.1. Diseño del puerto paralelo de E/S	56
3.2.2. Mapeado de E/S en la IBM PC	57
3.2.3. Sincronización	57
3.2.4. Acceso al bus y decodificación	59
3.2.5. Reloj / Contador programable 8253	61
3.2.5.1. Conexión del 8253	66
3.2.6. Conexión del ADC0809	70

3.3. Diseño del software	74
3.3.1. Funciones	76
4. Conclusiones	83
Bibliografía	85
Apéndices	
A Manual de usuario	87
B Hojas técnicas	105

Introducción.

La ciencia y la tecnología están estrechamente relacionados con el proceso de medición de variables. Es verdad que los instrumentos modernos de medición son uno de los frutos de la ciencia, pero también es cierto que sin la habilidad de medir, probablemente no habría ciencia. La lista es interminable, desde Tycho Brahe, Galileo y Newton; la ciencia tuvo necesidad de la experimentación y todos los experimentos implican mediciones.

La ciencia de la óptica por ejemplo, produjo las primeras grandes contribuciones a la instrumentación científica, como el telescopio y el microscopio.

El objetivo de la ciencia es descubrir las leyes de la naturaleza y como operan en sistemas complejos. El objetivo de la ingeniería es aplicar los descubrimientos de la ciencia a las necesidades humanas. Por eso, así como los científicos hacen descubrimientos que incrementan nuestro entendimiento del mundo que nos rodea, los ingenieros hacen inventos que intentan incrementar nuestra productividad (con ello nuestro nivel de vida), nuestra capacidad creativa y nuestra habilidad para sobrevivir.

En los últimos años, las computadoras han ganado popularidad en todas las áreas de trabajo, gracias a la versatilidad que pueden tener en cada aplicación, ya que es una herramienta poderosa en el desempeño de la tarea que se le encomienda y por tal motivo muchos sistemas se basan en éstas. A pesar de su enorme potencial de trabajo, estas máquinas pueden ser poco eficientes, a menos que tengan la adecuada programación de acuerdo a la aplicación que se les exijan.

Ahora bien, si el sistema desea interactuar con el mundo exterior se necesita una interfaz, la cual se puede definir como un sistema que une a dos o mas dispositivos, cuya función es tomar los datos del dispositivo fuente y producir niveles de corriente o de voltaje para alimentar al dispositivo destino.

Es en esta etapa donde los conocimientos acerca de los dispositivos electrónicos encuentran una aplicación importante.

El Analizador Dinámico de Señales (ADS) es un sistema electrónico hardware y software, el hardware se constituye de una tarjeta de adquisición de datos insertada en una computadora IBM PC o compatible (con microprocesador 286 o superior) y el software consta de un programa que simulará el funcionamiento de un Osciloscopio y de un Analizador de Espectros en la computadora; estos dos últimos son herramientas de medición primordiales en un Laboratorio de Electrónica.

Sus aplicaciones se basan en la utilización de dos canales de medición, con los que se pueden analizar señales de hasta 500 Hz de frecuencia, con un rango de voltaje de 20 Vpp. los resultados se muestran en el monitor de la computadora, de la misma forma que los muestra un Osciloscopio Digital, teniendo las mismas divisiones para analizar las magnitudes, ya sea de Voltaje o de Frecuencia. De esta forma podemos analizar señales con las relaciones Tiempo - Magnitud y Frecuencia - Magnitud.

El software se elaboró en lenguaje C++, contando con una interfaz gráfica y comandos para manipular la relación requerida, mostrando los valores de Voltaje y Frecuencia de cualquier punto de la señal que se esté analizando, así como otros comandos necesarios para el análisis de señales.

Justificación.

En la actualidad existen grandes avances en Instrumentación Electrónica, este avance incluye herramientas de medición sofisticadas y por lo tanto su valor es sumamente alto. En el estudio o análisis de señales, en este caso de Ingeniería Electrónica, es fundamental el uso de herramientas de análisis en el dominio del tiempo y en el de la frecuencia, es por esto que el uso de un Osciloscopio y el de un Analizador de Espectros debe ser una de las grandes ayudas para el pleno conocimiento de las señales. Con la existencia de computadoras en la mayoría de Laboratorios de Electrónica, podremos contar con tres herramientas de medición esenciales en un solo sistema, la computadora misma, un Osciloscopio y un Analizador de Espectros.

Además que cualquier persona puede construir su propia tarjeta de adquisición, cuyo análisis y diseño se trata en este trabajo de tesis, contando así con sus propias herramientas de medición.

Objetivo.

Lograr el diseño e implementación de un sistema de instrumentación de adquisición y procesamiento de datos para el uso en Laboratorios que no cuenten con instrumentos de medición esenciales, como son el Osciloscopio y el Analizador de Espectros. Las características principales de este diseño son:

- Construido a un bajo costo.
- Poder ser reproducido fácilmente.
- Proporcionar un instrumento de medición poderoso para el análisis de señales en un Laboratorio de Electrónica.

Contenido de la tesis.

El presente trabajo está organizado en cuatro capítulos principales, los cuales tratan lo siguiente:

Capítulo 1. Se dan las generalidades del trabajo así como una introducción y reseña de lo que es la creciente demanda de la computadora personal en el campo de la instrumentación.

Capítulo 2. Presenta el marco teórico para la comprensión de las herramientas matemáticas utilizadas.

Capítulo 3. Se presenta el diseño de la tarjeta de adquisición de datos, su interfaz con la computadora; además del hardware y software necesarios para poder direccionarla.

Capítulo 4. Se deducen las conclusiones, alcances y tendencias que se pueden lograr mediante una adecuada aplicación del Analizador Dinámico de Señales, además de un desarrollo futuro para lograr implementarlo a otros sistemas.

El apéndice A contempla un manual básico que presenta las bases para la configuración del hardware y los comandos para la utilización del software.

1

Consideraciones generales.

La medición de cantidades físicas data de hace mucho tiempo, pero la medición como un arte de precisión se ha registrado desde hace pocos años. Muchas de las cantidades que hoy medimos, no se conocían en tiempos anteriores.

En electrónica, las mediciones pueden ser de dos tipos; aquellas que se realizan a cantidades eléctricas tales como el voltaje, capacitancia o intensidad de campo y aquellas que implican cantidades físicas tales como la presión, temperatura o velocidad de un flujo¹. La instrumentación electrónica ha resuelto las necesidades de medición de la electrónica misma, pero su aplicación se esta adaptando rápidamente a otros campos.

¹ V. Grossman, Morris. *Test and measurement* , IEEE SPECTRUM, JANUARY 1993. pp. 55.

La medición exacta de dimensiones, temperatura, presión, potencia, voltaje, corriente, impedancia, propiedades de los materiales y demás variables físicas, es tan importante para la ingeniería como para la ciencia. Las mediciones exactas se necesitan para la eficiencia en el diseño en cualquier campo de aplicación, ya que la mayoría de las cantidades físicas se pueden convertir por medio de transductores a señales eléctricas, las cuales pueden ser amplificadas, filtradas, multiplexadas, muestreadas y medidas. Las mediciones se procesan fácilmente al convertirlas a una forma digital para realizar un análisis automático y poder almacenarlas para un uso posterior.

Los sistemas basados en computadoras personales son fundamentalmente una tarjeta de adquisición de datos respaldada por un software de aplicación apropiado, que permite que la computadora realice funciones de otros instrumentos electrónicos de medición. Con esto se ha incrementado el uso de las computadoras personales en la automatización de laboratorios, ya que se han podido implementar otros sistemas de medición muy importantes en los laboratorios de electrónica, como son el Osciloscopio y el Analizador de Espectros. Teniendo como resultado tres sistemas en uno, ahorro de espacio y la capacidad de usar los beneficios que la computadora implica.

Entre los instrumentos de medición de mayor importancia en el campo de la electrónica, se encuentran el Osciloscopio y el Analizador de Espectros, los cuales se encargan de analizar señales eléctricas en el dominio del tiempo y en el de la frecuencia, respectivamente.

Los instrumentos basados en computadoras personales operan en el dominio digital y requieren que las señales analógicas de entrada se conviertan a señales digitales, también requieren de un proceso para atenuar parte de la señal no deseada, como puede ser el ruido. Los procesos de digitalizar y procesar los datos toman un determinado tiempo, por lo que se puede considerar como una desventaja en comparación con los instrumentos que miden directamente en forma analógica. Sin embargo, los equipos de medición basados en computadoras personales tienen ventajas importantes como son la versatilidad, flexibilidad y pueden responder rápidamente a situaciones de prueba variables. Además de que se pueden conectar por medio de interfaces para formar un tablero o consola de instrumentación.

Una sola computadora puede soportar más instrumentos de prueba con la inserción de tarjetas de diferentes aplicaciones, que una mesa de trabajo llena de instrumentos convencionales. Con esto aumenta el espacio de trabajo y se proporciona una mayor eficiencia en el análisis y medición. Además de que los instrumentos basados en computadoras personales a diferencia de los instrumentos convencionales, pueden procesar los datos automáticamente y almacenarlos.

Existen dos tipos de tarjetas de adquisición de datos² :

- *Controlador/Interfaz*; las cuales se encargan de controlar un proceso por medio de una interfaz, ésta última se encarga de transportar las mediciones al controlador y regresar las instrucciones para realizar la acción necesaria para mantener el proceso; en la industria de la electrónica podemos encontrar una gran cantidad de aplicaciones para este tipo de tarjetas.

- *Procesamiento de señales Analógico/Digital*; este tipo de tarjetas encuentra su mayor aplicación en el campo del procesamiento digital de señales y en instrumentación digital de alta precisión. Se encarga de tomar los datos de señales analógicas y convertirlos a una forma digital, proporcionando así su salida a otros sistemas digitales.

1.1. **Procesamiento digital de señales.**

Las señales se usan para la comunicación entre humanos y a su vez éstos con máquinas, cuyo fin es el de investigar nuestro medio para descubrir detalles de estructura y de estado que son difíciles de percibir a simple vista. También se usan para controlar o utilizar su energía e información, el procesamiento de señales se refiere a la representación, transformación y manipulación de señales, conjuntamente con la información que contienen.

Por muchas décadas, el procesamiento de señales ha desempeñado un papel importante en diferentes campos como son; las comunicaciones, la ingeniería biomédica, la acústica, el radar, la sismología, la robótica, la electrónica de consumo y muchas otras. En la actualidad existen algoritmos muy sofisticados implementados en software y en hardware, los cuales se pueden encontrar en una gran cantidad de sistemas, desde sistemas altamente especializados de uso militar hasta sistemas electrónicos de consumo general. De forma rutinaria tratamos con estos sistemas, a diario observamos la televisión, escuchamos la radio en sistemas de alta fidelidad, etc.; el funcionamiento de estos sistemas depende del procesamiento de señales.

Anteriormente a los años 60's la tecnología del procesamiento de señales fue exclusivamente tecnología analógica en el tiempo continuo. La rápida evolución de computadoras y microprocesadores digitales junto a

² V. Byers, J. J.; *PC BASED TEST EQUIPMENT*, Electronics Now, October 1993, pp. 40

un importante desarrollo teórico causó el gran cambio de tecnologías analógicas a digitales, surgiendo con ello el procesamiento digital de señales, cuyo aspecto fundamental se basa en procesar una secuencia de muestras, las cuales se representan por medio de secuencias de números de precisión finita y el procesamiento se realiza por medio de cálculos digitales.

En la mayoría de casos en que interviene el procesamiento digital de señales, se realiza el procesamiento a una señal para obtener otra. Otra clase de problemas en el procesamiento digital de señales es la interpretación de la señal, por ejemplo, en un sistema de reconocimiento de voz, el objetivo es interpretar la señal de entrada o extraer su información.

Los sistemas para procesamiento digital de señales ha evolucionado a tal grado que podemos encontrar una gran cantidad de algoritmos implementados en software y en circuitos integrados, los cuales proporcionan grandes velocidades de procesamiento.

Desde la invención del cálculo en el siglo XVII, los científicos e ingenieros han desarrollado modelos para representar fenómenos físicos en términos de funciones de variables continuas y ecuaciones diferenciales. Desde entonces se han desarrollado técnicas numéricas para resolver estas ecuaciones ya que no existía forma analítica para resolverlas. Matemáticos del siglo XVIII como Euler, Bernoulli y Lagrange, desarrollaron métodos de integración numérica e interpolación de funciones de una variable continua. Fue hasta inicios de 1950 cuando surge el procesamiento de señales con el significado que ahora conocemos, pero solo en sistemas analógicos, los cuales se implementaron junto con dispositivos mecánicos en circuitos electrónicos.

Cuando surgen las computadoras digitales, solo eran accesibles en laboratorios científicos o de empresas de negocios, ya que tenían un precio alto y una capacidad limitada. Con la necesidad de un procesamiento de señales más sofisticado en algunas áreas de aplicación, se despertó un gran interés en el procesamiento de señales discreto en el tiempo. Una de las primeras aplicaciones de la computadora digital en el procesamiento de señales fue la investigación de prospección petrolera, donde se podían obtener datos, almacenarlos en medios magnéticos y después analizarlos.

Otra de las aplicaciones importantes fué el de simular sistemas de procesamiento de señales en una computadora digital antes de implementar el sistema en hardware analógico. Mas adelante, se logró obtener una aproximación o simulación de un sistema de procesamiento de señales analógico completo.

Como las señales eran procesadas en computadoras digitales, la investigación tendió a experimentar con algoritmos sofisticados para el procesamiento de señales, algunos de estos algoritmos eran poco flexibles para implementarse en la computadora digital y no encontraron una aplicación en sistemas analógicos. La evolución del procesamiento de señales discretas en el tiempo, se vio acelerada con la creación de algoritmos eficientes para la implementación de la transformada de Fourier, esta clase de algoritmos se conocen como la transformada rápida de Fourier (FFT); la FFT es importante por muchas razones, entre ellas, este algoritmo reduce el tiempo de procesamiento comparado con los que necesita el procesar con la transformada de Fourier, además de que permite la implementación de algoritmos para el procesamiento de señales más sofisticados que con el ahorro de tiempo de procesamiento mantiene una mayor interacción con el sistema.

Otro desarrollo importante en la historia del procesamiento de señales discretas en el tiempo, ocurrió en el campo de la microelectrónica, el surgimiento y la evolución de microprocesadores de propósito específico, ha reducido los costos de implementación de sistemas de procesamiento de señales discretas en el tiempo. Sin embargo, los primeros microprocesadores eran bastante lentos para poder implementarse en sistemas de tiempo real, a mediados de los 70's la tecnología de circuitos integrados avanzó a un nivel que permitió la implementación de microcomputadoras de punto fijo y de punto flotante a grandes velocidades, que junto a arquitecturas diseñadas especialmente han logrado implementar algoritmos para procesamiento digital de señales más sofisticados.

1.1.1.

Procesamiento digital de señales en instrumentación

Con la introducción del bus IEEE-488(*) a principios de los 70's, una nueva era surgió en la tecnología de la instrumentación. Antes, las mediciones se desarrollaban manualmente, lo cual era un trabajo lento y tedioso. Hoy en día, se pueden coleccionar rápidamente grandes cantidades de datos de

diferentes instrumentos de medición, además de poderse almacenar en una computadora³.

Inesperadamente, el problema cambió; de una falta de tiempo a ¿Qué hacer con esta gran cantidad de datos?, la respuesta se encontró en el procesamiento digital de señales.

Al mismo tiempo, el énfasis en los instrumentos de medición cambió de dispositivos analógicos a digitales. Los instrumentos de prueba analógicos inherentemente limitan los parámetros que se pueden medir a voltaje, corriente, frecuencia y fase. Los instrumentos digitales, por otro lado, derivan sus respuestas de algoritmos de procesamiento digital de señales que pueden medir parámetros no físicos y sofisticados, tales como los polos y ceros de funciones de transferencia de modelos complejos.

A este cambio a circuitos digitales vino como resultado el desarrollo de herramientas matemáticas para procesamiento digital de señales, por ejemplo, los analizadores de la transformada rápida de Fourier (FFT) y analizadores de microondas; los cuales ayudaron a interpretar la gran cantidad de datos por medio de instrumentos de medición más rápidos. En la sección 2.2.4., se analizará la FFT como una herramienta matemática para el procesamiento digital de señales.

Por supuesto, el cambio de instrumentación analógica a digital afectó las estructuras internas de los equipos de medición. Una configuración típica de instrumentación consiste en un generador de señales, un arreglo de adquisición de datos y una parte de procesamiento de datos, la cual extrae los parámetros de interés de los datos a analizar. Incomprensiblemente, la tecnología de los sensores y de los actuadores de la circuitería de la adquisición de datos dependen de la aplicación, mientras que la parte del procesamiento de datos solo se modifica y adapta a una aplicación específica, es decir esta parte es mucho más flexible.

Por otro lado, existen muchas restricciones impuestas por la naturaleza que tienen un gran efecto en los resultados del análisis real de los datos. Por ejemplo, una restricción es el requerimiento de un intervalo finito de observación. Teóricamente, una señal puede evaluarse en un intervalo infinito, en el que se puede realizar el cambio del dominio del tiempo al de la frecuencia y viceversa. Sin embargo, en la práctica se asume que el comportamiento de una señal en un intervalo que va de milisegundos a

(*) Conocido como el Bus de interfaz de propósito general, (GPIB, General Purpose Interface Bus) , que permite conectar hasta 15 instrumentos o dispositivos para comunicarse entre ellos.

³ V. Oliver and Cage; *Electronic measurements and Instrumentation*, Inter-University Electronics, Series 1971.

un minuto, se puede analizar y adquirir las características fundamentales para analizar con exactitud la forma de la señal.

Con la anterior formulación, se ha venido basando el análisis de señales; mas aún ahora con la aplicación de la computadora en el campo del procesamiento digital de señales, se hace más necesario el uso de técnicas digitales. Lo anterior ha acelerado el desarrollo de instrumentación digital para el procesamiento de señales. Los datos se convierten a una forma digital para facilitar las operaciones que normalmente se requieren para el análisis de señales. Además de que fácilmente se pueden implementar herramientas matemáticas, las cuales necesitan de operaciones algebraicas que son muy difíciles de implementar con señales analógicas.

Cabe advertir que las técnicas de análisis de señales pueden reducir o alterar los datos, pero obviamente no pueden proporcionar nueva información⁴. Lo anterior debido a que la misma naturaleza impone restricciones que causan efectos en los resultados del análisis de los datos reales, una de estas restricciones es, la que solo podemos analizar cierta señal en un tiempo finito, ya que no se cuenta con suficiente capacidad de almacenamiento para poder almacenar toda la información de una señal, ya que estaremos hablando de un análisis de tiempo infinito; es por lo anterior, que las herramientas de análisis de señales solo obtienen información de la señal al analizarla en un intervalo de tiempo finito. Además de la restricción anterior, las mismas herramientas tienen sus propias restricciones, con lo cual la información obtenida es todavía menor. Esta información obtenida es una parte de la información total, sin embargo nos ayuda a conocer o predecir la información restante de la señal.

Finalmente, podemos tener la idea de qué tan importante ha resultado el procesamiento digital de señales, ya que es una de las partes que se han desarrollado de una forma impresionante para analizar una gran cantidad de datos, además de que ayuda a disminuir o controlar parte de la señal no deseada, por ejemplo el ruido.

Como se ha visto anteriormente, con el uso de herramientas digitales, el procesamiento digital de señales vino a ser el complemento ideal para lograr sistemas más sofisticados, con mayores ventajas y aplicaciones.

Todas las señales electrónicas, de todos los fenómenos pueden describirse como una función del tiempo o de la frecuencia. Cuando un fenómeno es cíclico, teniendo una periodicidad definida, la relación básica entre la frecuencia y el tiempo es simplemente el inverso del otro. El Osciloscopio

⁴ V. Potter, Ronald W.; *SIGNAL ANALYSIS BY DIGITAL TECHNIQUES*, Hewlett Packard Company. 1987, pp. 107

muestra la señal en una relación Magnitud-Tiempo, es decir mide la magnitud de la señal con respecto al tiempo; mientras que el Analizador de Espectros muestra la señal en una relación Magnitud-Frecuencia. Por lo anterior podemos ver la importancia de contar con estas dos herramientas en el análisis de señales.

La figura 1.1 muestra un ejemplo de señales representadas en el dominio del tiempo y en el de la frecuencia. Cuando dos señales diferentes f_a y f_b , las cuales se muestran en el dominio del tiempo (relación Magnitud-Tiempo) por la figura 1.1 (a), se aplican simultáneamente a la entrada de un Analizador de Espectros, se podrán ver sus componentes en frecuencia (relación Magnitud-Frecuencia), mostradas como líneas verticales en 200kHz (f_a) y 300kHz (f_b); figura 1.1 (b).

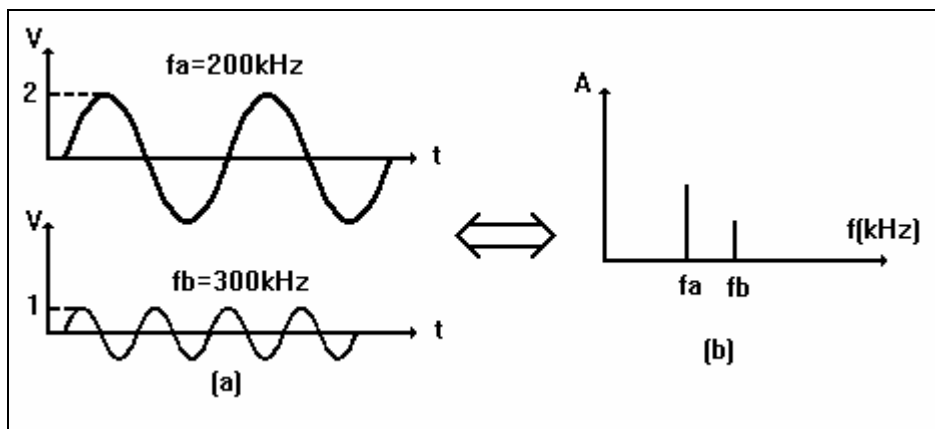


Figura 1.1 Representación de una señal en (a) el dominio del tiempo y (b) el dominio de la frecuencia.

Así como el Osciloscopio es un instrumento cuya función básica es mostrar las características del fenómeno en el dominio del tiempo, también el Analizador de Espectros es un instrumento cuya función es mostrar estas características en el dominio de la frecuencia. Se debe reconocer que las dos descripciones del mismo fenómeno (el dominio del tiempo para el Osciloscopio y el dominio de la frecuencia para el Analizador de Espectros) no son independientes. Es decir, si alguna de las dos se conoce, por medio de herramientas matemáticas tales como la serie y la transformada de Fourier, podemos conocer a la otra.

Este es el caso del presente trabajo, ya que con el diseño de la tarjeta de adquisición de datos estaremos básicamente realizando la función del Osciloscopio Digital, pero para el funcionamiento del Analizador de Espectros, se implementará la transformada rápida de Fourier en software

para obtener las características del dominio de la frecuencia a partir del dominio del tiempo de la señal.

El corazón del Osciloscopio basado en una computadora personal es el convertidor analógico a digital. El número de bits de este determina la resolución de la señal muestreada. Algunas tarjetas de adquisición de datos cuentan con una memoria RAM, en este caso se usa la memoria de la PC, es decir, la tarjeta captura la forma de onda y la envía a la memoria de la PC; aquí es donde los datos se procesan y se muestran en el monitor de la computadora. La cantidad de memoria de la computadora determina la cantidad de muestras de la forma de onda que pueden capturarse.

El otro instrumento es el Analizador de Espectros, el cual examina el dominio de la frecuencia de los datos de entrada. En este caso, las mediciones en el dominio de la frecuencia se hacen con ayuda de la transformada rápida de Fourier (FFT), la cual procesa de forma digital una señal en un periodo de tiempo específico para proveer información de la frecuencia, amplitud y fase de la señal muestreada. Pueden analizarse señales periódicas y no periódicas.

En este caso, el Analizador de Espectros con la ayuda de la FFT procesa los datos capturados por el Osciloscopio Digital (por la tarjeta de adquisición) a través de software. Los cálculos de la FFT se hacen en el CPU de la computadora, para obtener mayor velocidad de procesamiento.

1.2. Instrumentación virtual.

Con el desarrollo de la PC y de las herramientas antes mencionadas, en el mundo de la instrumentación se ha venido usando el término "Instrumentación Virtual"; este término describe la combinación entre instrumentos de programación y PC's de propósito general. Con este término, ha surgido una nueva generación de herramientas de instrumentación. Con los instrumentos virtuales, se puede utilizar una gran variedad de instrumentos hardware en un sistema, aumentando su flexibilidad y potencial de trabajo con ayuda de herramientas software de alto nivel con aplicaciones gráficas.

Con las nuevas técnicas de control en la instrumentación tales como, programación basada en registros y memoria compartida, se obtiene un máximo rendimiento en el uso de la computadora para adquisición, análisis y presentación de datos.

En el contexto de electrónica, los instrumentos tradicionales se definen como una caja inmóvil, a la cual se conectan señales de entrada y muestran como salida el análisis de la entrada. Se puede generalizar la funcionalidad de un instrumento tradicional en tres elementos, adquisición de datos, análisis de los datos y su presentación. En general, la instrumentación tradicional contiene estos tres elementos como se muestra en la figura 1.2. Un instrumento virtual también contiene estos tres elementos. Sin embargo, en los instrumentos virtuales estos tres elementos no residen en una misma caja o lugar. El uso del termino "virtual" refleja el hecho de que estos elementos no necesariamente deben estar en la misma caja o pertenecer a una misma unidad. La revolución de las PC's ha proporcionado a los usuarios una gran cantidad de herramientas de procesamiento y grandes capacidades de presentación de los resultados. Si se comparan las capacidades ya establecidas de un instrumento tradicional con la flexibilidad que el usuario puede obtener en una computadora, se podrá tener como resultado un instrumento virtual. La computadora puede funcionar en cualquiera de las tres áreas, adquisición de datos, análisis y/o presentación.

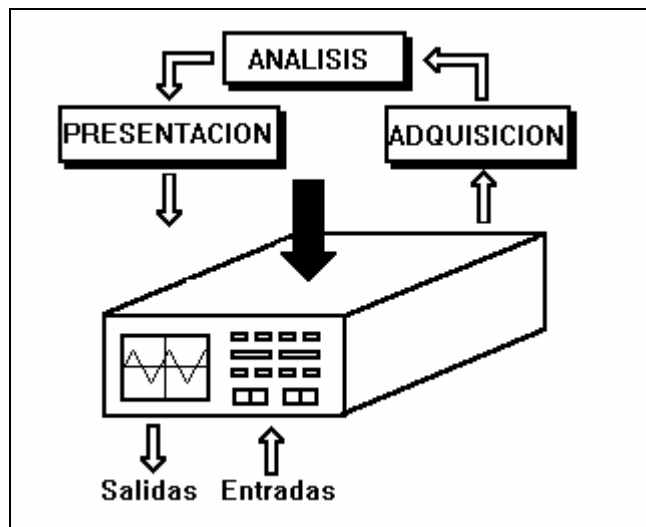


Figura 1.2. Elementos de un instrumento tradicional.

La definición de instrumento virtual ha tomado diferentes significados, pero la que más comúnmente se utiliza es la de "La utilización de una capa de software y de hardware incorporada a una computadora de propósito general, en la cual el usuario puede interactuar con la

computadora como si en verdad trabajara con un instrumento electrónico tradicional”⁵.

1.2.1.

Fases de diseño de un instrumento virtual.

Para el diseño de un instrumento virtual, se han normalizado tres fases:

1. La simulación de la funcionalidad de los instrumentos tradicionales en una computadora.
2. Análisis de la arquitectura de los instrumentos.
3. El reconocimiento y adopción del desarrollo de estructuras en instrumentación virtual.

Fase 1.- **La simulación de la funcionalidad de los instrumentos tradicionales en una computadora.** Desde la estandarización del bus IEEE 488 (GPIB), se han podido controlar instrumentos por medio de la computadora; esto ha tenido un gran desarrollo con la evolución de la PC. En los últimos 10 años, la computadora ha controlado a la gran mayoría de instrumentos, con ello han surgido herramientas de software para hacer más fácil el uso de estos instrumentos tradicionales. Se ha obtenido un gran incremento de librerías de análisis de datos, de gráficos, de manejadores de instrumentación para hacer más fácil el diseño de interfaces.

Se puede obtener o diseñar fácilmente cualquier herramienta software para ampliar la funcionalidad de los sistemas de instrumentación, con la capacidad de crear su propio análisis y presentación de datos para cualquier sistema de instrumentación. Esta capacidad también incluye el uso de instrumentos tradicionales, los cuales se pueden conectar a la computadora, vía GPIB o RS-232. Se pueden conectar cientos de instrumentos independientes a las PC's para crear instrumentos virtuales más poderosos.

Fase 2.- **Análisis de la arquitectura de los instrumentos.** Como respuesta al desarrollo de la instrumentación virtual, han surgido dos nuevas técnicas de instrumentación hardware, la inserción en la PC de tarjetas de adquisición e instrumentación por medio del bus VXI. Estas nuevas técnicas son el resultado de la evolución de los componentes usados en la construcción de instrumentos tradicionales que se han acoplado a la funcionalidad de la PC. Hasta el momento, no ha surgido una herramienta que combine elementos hardware y software que

⁵ V. National Instruments, *DATA ACQUISITION SEMINAR*, January 1995 Edition, pp.8

desarrollen una adquisición, control, análisis y presentación de los datos de forma tan eficiente como lo hace la PC.

El diseño de la instrumentación tradicional ha tratado de lograr los máximos avances de la tecnología, pero la arquitectura interna de estos ha tendido a tomar los estándares de la computadora, por ejemplo, ahora es común que un instrumento tradicional incluya un microprocesador, un diseño de memoria y librerías de software para la adquisición, análisis y presentación de los datos; es por esto, que la computadora es la plataforma ideal para la construcción de instrumentos virtuales.

La instrumentación, en cuanto a hardware se refiere, incluye a convertidores analógico a digital (ADC), digital a analógico (DAC), componentes de sincronización de entrada / salida (TIO), componentes de tiempo y de acondicionamiento de señales. Estos componentes se han convertido en elementos modulares, los cuales se pueden obtener por separado, insertarse en una tarjeta principal que se conecta directamente al bus de la computadora, obteniendo así instrumentos a la medida y a las necesidades que se requieran. Como se sabe, las PC's actuales cuentan con slots de expansión, en los cuales se pueden instalar tarjetas modulares y conectarlas directamente al bus del microprocesador. En cuanto a software, las herramientas software se cargan en la memoria ROM interna. Este software es único para cada instrumento, pero su código esta compuesto de módulos en forma de librerías, las cuales pueden ser utilizadas para diseñar otros instrumentos. En instrumentación virtual, el software para un instrumento en particular también usa partes reusables del código, pero a diferencia de los instrumentos tradicionales, en los que el software esta limitado a ser usado en una ROM específica; la funcionalidad de un instrumento virtual, permite que el software se almacene en archivos en un disco duro o flexible que puede fácilmente ser instalado en la computadora. Con esto se hace más flexible la utilización de varios instrumentos en la misma computadora.

Una desventaja, es que contando con varios instrumentos tradicionales, podemos tener la representación de los datos en cada uno de ellos al mismo tiempo, mientras que usando instrumentación virtual, el número de instrumentos virtuales se restringe a el uso de la pantalla de la computadora, sin embargo se pueden ver todas las presentaciones que queramos, solo que no al mismo tiempo que teniendo instrumentos tradicionales. Esta desventaja se reduce al contemplar las ventajas de flexibilidad y facilidad de manipulación que ofrece la instrumentación virtual.

Para analizar la arquitectura de los instrumentos tradicionales y crear un nuevo aprovechamiento para la construcción de instrumentos al

compartir recursos de sistema, los componentes claves son; el bus del procesador y la sincronización entre la interfaz y la computadora.

El bus del procesador es la base para realizar un análisis y presentación de datos en tiempo real. Los componentes internos de la interfaz se conectan directamente al bus y todos los registros hardware son mapeados en memoria por el procesador. El procesador, por lo tanto, puede controlar directamente a los componentes hardware de la interfaz; de forma que la interfaz puede llamar por medio de una interrupción al procesador cuando sean necesarios sus servicios. Se pueden transferir datos en forma binaria o en cualquier otro formato, en forma paralela y a velocidades de transmisión muy altas. Se puede tener un acceso directo a memoria (DMA) y técnicas de acceso al bus (buffering) para analizar o preparar los datos mientras se adquieren o envían los resultados a un archivo o a una impresora.

La otra clave es la de sincronizar los tiempos entre la interfaz y la computadora, de esto también depende la adquisición y generación de datos en tiempo real. Es de gran importancia esta etapa, ya que como se puede advertir, en un instrumento tradicional, la base de tiempos es la misma, por lo que no existen problemas de sincronización, ahora bien, en un instrumento virtual, el problema es el de sincronizar los tiempos entre la computadora y la interfaz o el instrumento. En el presente trabajo, se toma la base de tiempo de la computadora para ser usada por la interfaz.

Fase 3.- En la primera fase se comparan las capacidades de la instrumentación virtual con respecto a la instrumentación tradicional. En la segunda fase se analiza la arquitectura de las herramientas de instrumentación y la creación de nuevas técnicas de construcción de instrumentos. La tercera fase se conoce como: **El reconocimiento y adopción del desarrollo de estructuras en instrumentación virtual.**

Con la revolución de las computadoras, el desarrollo de lenguajes de alto nivel y lenguajes orientados a objetos, se ha venido incrementando el desarrollo de los instrumentos virtuales en todos los campos de la ciencia. En resumen, una estructura de software para un instrumento virtual es la parte principal para la automatización de adquisición de datos y el control de sistemas de instrumentación. Estas estructuras combinan manejadores de bajo nivel hardware y aplicaciones software de alto nivel para manipulación de señales. Con esto, se cuenta con una variedad de lenguajes de programación que el usuario puede elegir dependiendo de sus habilidades y sus necesidades. En este caso el lenguaje que se ha elegido es el lenguaje C++. En el cual se realiza la programación para el análisis y presentación de los datos.

Como se ha mencionado, en los últimos años el gran desarrollo de herramientas para la implementación de instrumentos virtuales, ha

demostrado la gran capacidad de estos instrumentos, por lo que se han venido desarrollando al mismo tiempo herramientas más poderosas en software y con respecto a hardware los componentes electrónicos se han venido especializando y modularizando para poder tomar el ritmo de desarrollo en este campo de acción. Los resultados son en verdad sorprendentes y muestran un gran potencial en su aplicación, es por esto que cada vez es más común encontrarse con instrumentos virtuales en cualquier campo de la ciencia.

1.2.2.

Elementos de un instrumento virtual.

Hasta ahora se ha descrito el funcionamiento de un instrumento tradicional, el cual consiste de tres componentes esenciales, la adquisición, análisis y presentación. Un instrumento virtual también cuenta con estos componentes, sin embargo, estos componentes presentan una gran flexibilidad y son fácilmente maniobrables.

En el mercado podemos encontrar una gran cantidad de hardware especializado, es decir en forma de módulos para poder adaptarlos a la aplicación que deseamos. Estos módulos pueden ser fácilmente intercambiados, configurados y hasta diseñados, ya que intuitivamente podemos apreciar que es más fácil diseñar un módulo en particular que un sistema en general.

El análisis de datos se puede realizar en la computadora y puede consistir en cualquier tipo de análisis que se desee y puede estandarizarse o adaptarse a las necesidades de una aplicación en particular. Una de las ventajas principales es la de la presentación de los datos, éstos se manipulan y se adecúan en la computadora. La presentación se ha realizado tomando en cuenta que esta sea lo más similar a la que proporciona un instrumento tradicional, solo que con una flexibilidad sorprendente y múltiples paneles dinámicos, dando como resultado, un instrumento fácil de usar para obtener su mayor provecho.

Uno de los beneficios más notorios de la instrumentación virtual es que se pueden implementar varios instrumentos en un mismo panel y cambiar de un instrumento a otro con un comando de software. También se pueden visualizar varios instrumentos al mismo tiempo en la pantalla en forma de ventanas independientes, con lo que se tiene un mayor control sobre el sistema. Este método optimiza el uso de equipo, ya que se puede utilizar el hardware para una gran variedad de funciones, por ejemplo, teniendo varios instrumentos tradicionales, es necesario que cada uno de ellos cuente con sus elementos independientes, mientras que en un

instrumento virtual, estos elementos pueden ser reutilizados por otros elementos, optimizando de alguna forma el uso del equipo.

En la figura 1.3 se presenta un diagrama en el cual se muestran los elementos que conforman a un instrumento virtual.

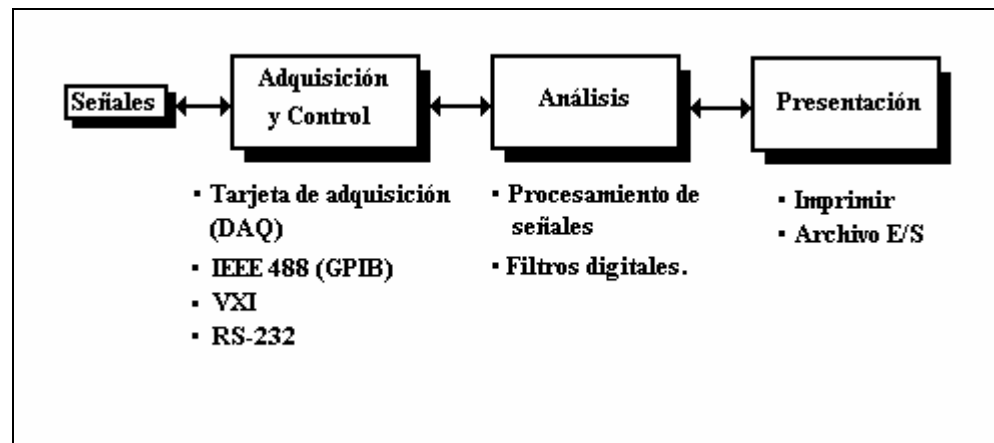


Figura 1.3 Esquema general de un sistema de instrumentación virtual.

La adquisición de datos como definición se ha venido realizando desde que la ciencia experimental necesito analizar el comportamiento de ciertos fenómenos de la naturaleza. El análisis que se realizaba en aquella época era del todo analógico y los resultados se plasmaban en tablas de control.

La adquisición de datos ha complementado el objetivo de la medición, ya que ha proporcionado la flexibilidad de interactuar con otros sistemas que permiten almacenar los resultados obtenidos para un procesamiento posterior.

La ayuda de sistemas digitales de gran capacidad, como es la computadora personal, el gran avance que ha tenido la electrónica en circuitería de medición y en procesamiento digital de señales, ha facilitado el procesamiento y análisis de señales analógicas.

La etapa que realiza la interacción entre estos tipos de señales es la conversión analógica a digital. Como resultado se han obtenido sistemas de adquisición de datos que ofrecen gran exactitud, velocidad y suficiente capacidad de almacenamiento. En el mercado podemos encontrar una gran variedad de sistemas de adquisición de datos para diferentes aplicaciones.

Más adelante se dan las bases para entender la funcionalidad de un convertidor analógico a digital y se analizará la importancia que tiene en el presente proyecto.

Haciendo referencia a la figura 1.3, podemos analizar sus bloques de la siguiente forma:

El primer bloque que es el de señales, que en sí es la etapa donde se adquieren y miden las señales del mundo real (en su mayoría analógicas) por medio de un transductor o sensor; la siguiente etapa se encarga de convertir la forma de la señal a una digital, como entrada a la siguiente etapa de análisis de los datos.

En general un transductor o sensor mide un parámetro físico, tal como presión, temperatura, vibración, etc. y lo convierte a una corriente o voltaje eléctrico. Es aquí donde empieza la labor de los circuitos electrónicos, los cuales procesan la señal obtenida por el transductor y proporcionan una salida a un sistema digital destino. Con esta etapa se completa el procesamiento analógico. El análisis de esta etapa se lleva a cabo en la sección 3.1.

Después de que se termina el procesamiento analógico, la señal se convierte a forma digital por medio de un convertidor analógico / digital o A/D, para poder alimentar a una gran variedad de posibles sistemas digitales, por ejemplo : una computadora personal, un controlador digital o un transmisor de datos digital.

En este caso, el sistema de adquisición de datos alimenta a una computadora personal, para que ésta simule el funcionamiento de un Osciloscopio y el de un Analizador de Espectros, realizando el análisis de señales en el dominio del tiempo y en el de la frecuencia, respectivamente. Es necesario recordar que el Analizador Dinámico de Señales es tan solo una aplicación, necesaria en un laboratorio de electrónica y más aún en el amplio campo del procesamiento digital de señales.

La tercera etapa, trata el análisis de los datos, en la cual los datos ya en forma digital y en forma ordenada son procesados o analizados con herramientas matemáticas o de procesamiento de señales. Esta es la etapa de más importancia para la ingeniería, ya que en base a la innovación de nuevas herramientas matemáticas y digitales, se ha logrado un gran avance. Por ejemplo, en el mercado podemos encontrar microprocesadores dedicados a realizar procesamiento digital de señales especializados. En el capítulo dos se analiza esta etapa.

La cuarta y última etapa, es la de la representación de los datos; en la cual se muestra el resultado de la manipulación o procesamiento que se

le ha realizado a los datos originales. Desde el punto de vista económico, este es el punto más importante en cuanto a diseño, ya que con una buena presentación e imagen, podemos llamar más la atención hacia el sistema de adquisición de datos. La implementación de esta etapa en el ADS se explica en la sección 3.3.

En este caso las dos últimas etapas se implementan en base a una computadora personal IBM PC o compatible; la segunda etapa se realiza por medio del bus, memoria, dispositivos de almacenamiento y el procesador de la computadora; la tercera etapa se realiza con herramientas matemáticas y de software para realizar funciones de procesamiento digital de señales. La cuarta etapa también se realiza con ayuda de la computadora, en la cual interviene en gran parte el software, dispositivos de almacenamiento y de entrada/salida, de los cuales el monitor es el más utilizado para presentar los datos de salida.

2

Análisis matemático de señales.

Como podemos apreciar en la figura 1.3 la tercera etapa es la de análisis de los datos, ya que las etapas anteriores proporcionan la información de la señal ya acondicionada para su respectivo análisis; esta etapa de análisis se encarga de preparar los datos para su presentación final, entendiendo lo de preparar, como el hecho de adaptar o modificar los datos dependiendo del análisis o la aplicación. En este caso esta etapa consiste en adaptar los datos de la señal en dos arreglos, el primero contendrá a la señal en el dominio del tiempo y el segundo contendrá el espectro de la señal después de haberle realizado el cálculo de la FFT. Estos datos al final de la etapa deben estar disponibles para poder ser mostrados gráficamente en el monitor de la computadora.

A continuación se analizará la tercer etapa, que es la de análisis.

En el diseño de la tarjeta de adquisición, que se encarga de realizar el funcionamiento del Osciloscopio, es importante la velocidad de muestreo, ya que de ello depende la capacidad de análisis del sistema, por esto es importante conocer el teorema de muestreo. Para simular el funcionamiento del Analizador de Espectros, con la ayuda de la tarjeta de adquisición podemos muestrear la señal y almacenarla en la memoria de la PC, es decir la señal en la memoria es en el dominio del tiempo y con la implementación mediante software del algoritmo de la FFT podemos obtener el espectro de la señal.

A continuación se mencionan las herramientas matemáticas que utiliza el ADS y la teoría de operación de un convertidor analógico a digital por aproximaciones sucesivas, siendo éste último la base del diseño de la tarjeta de adquisición.

2.1.

Teorema de muestreo.

Bajo ciertas condiciones, una señal de tiempo continuo se puede representar y reconstruir completamente, partiendo del conocimiento de sus valores instantáneos, o muestras, igualmente espaciadas en el tiempo. Esta propiedad se deriva de un resultado básico conocido como el teorema de muestreo.

Gran parte de la importancia del teorema de muestreo reside en su papel de puente entre las señales de tiempo continuo y las señales de tiempo discreto.

En este caso, la entrada a nuestro sistema es la salida de un transductor o sensor, el cual proporciona una señal analógica, es decir una señal de tiempo continuo; como sabemos al interactuar con un sistema digital, es necesario convertir esta señal a tiempo discreto. Ahora bien, el caso ideal es el de procesar todos los valores definidos de la variable analógica, sin embargo, debido a las limitaciones de almacenamiento que la computadora representa, debemos limitar el número de puntos de la muestra. Estos valores, se toman a instantes discretos de tiempo, los cuales constituyen las "muestras" de la señal analógica. La razón a la cual se muestrea la señal determina la exactitud de la representación como una función discreta en el tiempo.

La exactitud de la razón de muestreo y su determinación se analizan por medio del teorema de muestreo, que se define como:

Dada una señal $x(t)$ limitada en banda con $X(\omega)=0$ para $|\omega|>\omega_m$, siendo ω_m la frecuencia de muestreo, figura 2.1(b). Entonces $x(t)$ está determinada unívocamente por sus muestras $x(nt)$, $n=0, \pm 1, \pm 2, \dots$ si

$$\omega_s > 2\omega_m$$

donde

$$\omega_s = 2\pi/T$$

Dadas estas muestras, podemos reconstruir $x(t)$ generando un tren de impulsos periódicos en el que los impulsos sucesivos tienen amplitudes que corresponden a valores de muestras sucesivas. Este tren de impulsos es entonces procesado a través de un filtro paso bajas con ganancia T y cuya frecuencia de corte es mayor que ω_m y menor que $(\omega_s - \omega_m)$. La salida resultante será exactamente igual a $x(t)$.

En forma matemática tenemos:

$$x_p(t) = x(t)p(t)$$

$$p(t) = \sum_{n=-\infty}^{\infty} \delta(t - nT)$$

$$x_p(t) = \sum_{n=-\infty}^{\infty} x(nT)\delta(t - nT)$$

$$\omega_s = \frac{2\pi}{T}$$

donde T es el periodo de muestreo y ω_s es la frecuencia fundamental. La siguiente figura muestra gráficamente este proceso.

La figura 2.2(a) representa el espectro de frecuencia de la señal $x(t)$ mostrada en la figura 2.1 (b). Como podemos ver, $X(\omega)=0$ para frecuencias mayores a ω_m ; es decir, que $x(t)$ no tiene frecuencias fuera de aquellas que corresponden a ω_m ; se dice entonces que $x(t)$ es una señal limitada en banda. El espectro para el tren de impulsos espaciados en el tiempo por T [figura 2.1 (c)] es también una secuencia de impulsos separados por $\omega_s=2\pi/T$, que es la frecuencia de muestreo [figura 2.2(b)].

Para recuperar la señal analógica $x(t)$ de su representación de muestras debemos recuperar su espectro totalmente en el dominio de la frecuencia. Un filtro paso bajo, cuya respuesta en frecuencia se indica por las líneas punteadas de la figura 2.2(c), complementa la recuperación de $x(t)$; Como veremos a continuación, entre mayor sea ω_s respecto a ω_m la recuperación será más fiable. Matemáticamente se debe cumplir que :

$$T > \frac{1}{2\omega_m}$$

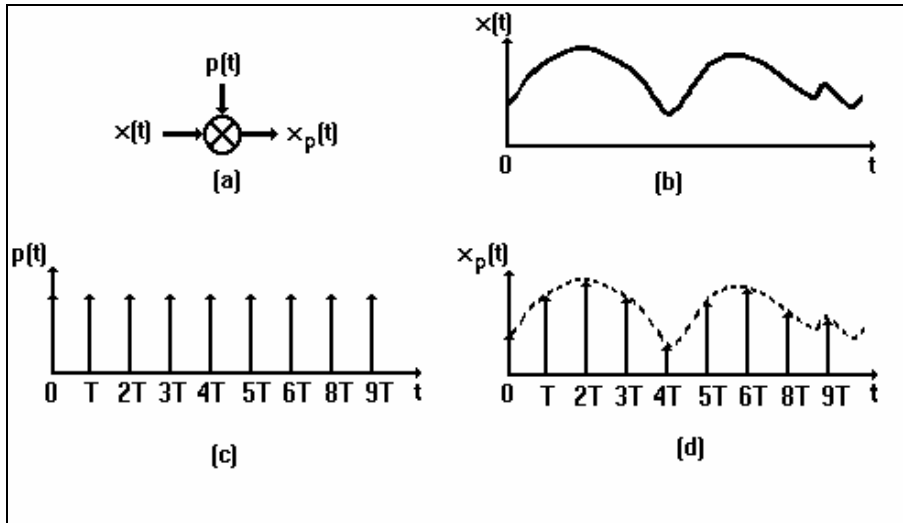


Figura 2.1 Teorema de muestreo. (a) diagrama de flujo, (b) señal moduladora, (c) señal portadora y (d) señal modulada.

Lo anterior muestra que es bastante fácil recuperar una señal muestreada. Sin embargo, existe la hipótesis de que la señal sea limitada en banda y que la frecuencia de muestreo ω_s , es no solamente más grande que ω_m , de hecho $(\omega_s - \omega_m)$ es más grande que ω_m [figura 2.2(c)]. De esta forma:

$$(\omega_s - \omega_m) > \omega_m \rightarrow \omega_s > 2\omega_m$$

o ω_s debe ser el doble o más que ω_m . En forma análoga al muestreo en el dominio del tiempo, existe un teorema de muestreo en el dominio de la frecuencia.

La figura 2.3 muestra una señal real limitada por el uso de un filtro paso bajo antes del muestreo. La información contenida en las frecuencias más altas se desprecian, dando así lugar a lo que se conoce como "error de omisión"⁷. Todo el procedimiento se realiza en una señal que no es la original. Si la información perdida o despreciada por el filtrado no es importante, este procedimiento es tan válido mientras que ω_s sea más grande que el doble de ω_m .

⁷ V. Oppenheim, Alan, W., Schaffer, Ronald W; *Discrete-Time Signal Processing*, Prentice Hall, 1989, pp. 84

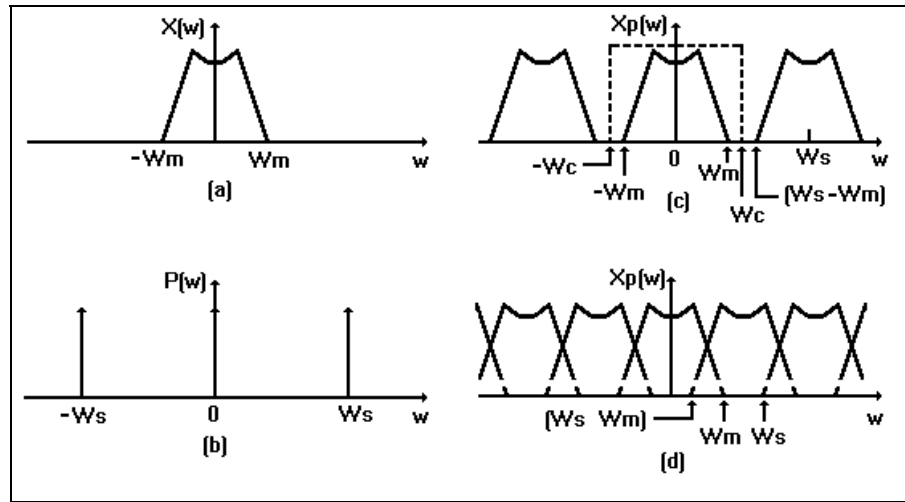


Figura 2.2 Representación en el dominio de la frecuencia de las señales de la figura 2.1

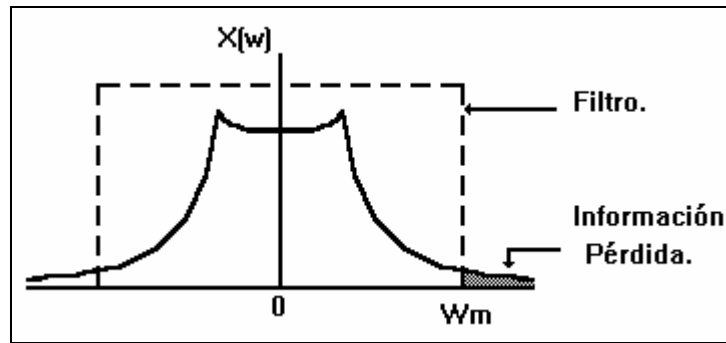


Figura 2.3 Espectro de la señal moduladora $x(t)$ real.

Cuando la frecuencia ω_s es menor que el doble de la frecuencia más alta de la señal, el espectro de $x(t)$ no se puede distinguir en $X_p(\omega)$ y por lo tanto no se puede recuperar satisfactoriamente por el filtro paso bajo. La figura 2.2(d) muestra el traslape que ocurre al presentarse esta situación, conocida como "aliasing".

Para un mejor entendimiento de este efecto, consideremos una onda cosenoidal ideal con frecuencia ω_0 :

$$x(t) = \cos 2\pi\omega_0 t$$

cuyo espectro se presenta en la figura 2.4(a).

Si muestreamos esta forma de onda con un tren de impulsos a una frecuencia ω_s , el espectro será una serie infinita de pares de impulsos

como muestra la figura 2.4(b) (para mayor claridad en la gráfica se ha marcada cada par de impulsos correspondientes con un número igual de ventanas). Las frecuencias negativas son consecuencia matemática del proceso de modulación, y nos permite centrar la atención en las frecuencias positivas.

En la figura 2.4(b) la frecuencia de muestreo es $\omega_s > 2\omega_0$. Un filtro paso bajo, cuya respuesta en frecuencia se indica en la misma figura, puede recobrar completamente el espectro de la onda cosenoidal.

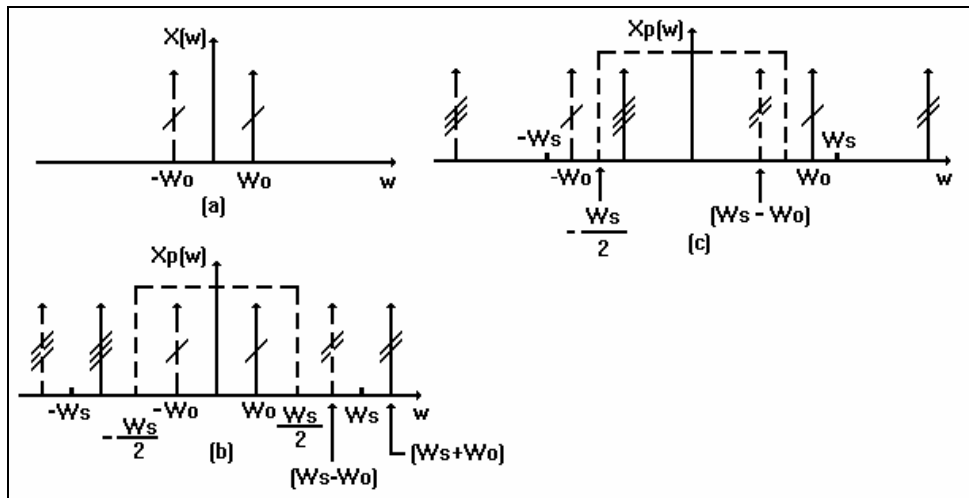


Figura 2.4 "Aliasing". (a) espectro de la señal original, (b) espectro de la señal muestreada y (c) espectro de la señal a frecuencias de muestreo bajas.

Ahora analizaremos qué sucede a frecuencias de muestreo más bajas, para $\omega_s < 2\omega_0$. La figura 2.4(c), permite mostrar de forma clara cómo al disminuir la frecuencia de muestreo, resulta que $(\omega_s - \omega_0) < \omega_0$.

Analizando un par de impulsos generados por el proceso de modulación no muestra que la señal que queremos recuperar por el filtro paso bajo, sea una onda cosenoidal, con una frecuencia $(\omega_s - \omega_0)$. Se hace notar que el único cambio que se introdujo en la figura 2.4(c) es una proporción más baja entre ω_s y ω_0 , a comparación de la figura 2.4(b).

Los resultados se pueden resumir al mencionar que cuando ocurre el efecto de aliasing, la frecuencia original ω_0 , a frecuencias menores a $(\omega_s - \omega_0)$ sufre cambios.

2.2. Análisis de Fourier.

El desarrollo de las técnicas de análisis de Fourier tiene una larga historia que involucra a un gran número de personas, entre ellas Euler, D. Bernoulli, J. L. Lagrange, S.F. Lacroix, G. Monge, P. S. Laplace y por supuesto el matemático y físico teórico francés Jean Baptiste Joseph Fourier (1768 - 1830), quien dijo que cualquier señal en el dominio del tiempo está compuesta de una suma de senoidales, cada una de las cuales tiene su propia frecuencia, amplitud y fase.



Jean Baptiste Joseph Fourier (1768-1830)

Al hablar de senoidales, se manifiesta el uso de señales periódicas básicas como son las exponenciales complejas, éstas son de la forma e^{st} , donde s es un número complejo de la forma $s=\alpha+j\beta$.

Una de las ventajas de usar exponenciales complejas es que la respuesta de cualquier sistema lineal e invariante en el tiempo a una entrada exponencial compleja es la misma exponencial compleja modificada solo en amplitud, esto es, las exponenciales complejas son autofunciones de los sistemas lineales.

Para el desarrollo matemático de las herramientas de Fourier utilizadas en este trabajo, consultar la referencia bibliográfica⁸.

2.2.1.

⁸ V. Oppenheim, Alan V., Willsky, Alan S.; SEÑALES Y SISTEMAS, Prentice Hall, 1994, cap. 4 y 5.

Series de Fourier.

Se dice que una señal es periódica si para algún valor positivo de T , diferente de cero:

$$x(t)=x(t+T) \text{ para toda } t \quad (2-1)$$

el período fundamental T_0 de $x(t)$ es el mínimo valor positivo de T , diferente de cero para el que se cumple la ecuación (2-1), el valor $2\pi/T_0$ se conoce como la frecuencia fundamental.

Como se ha mencionado anteriormente, las dos señales periódicas básicas que utilizaremos son, la cosenoide

$$x(t) = \cos \omega_0 t$$

y la exponencial compleja periódica

$$x(t) = e^{j\omega_0 t} \quad (2-2)$$

Estas dos señales son periódicas con frecuencia fundamental ω_0 y período fundamental $T_0=2\pi/\omega_0$. Asociado con la señal de la ecuación (2-2) se encuentra el conjunto de exponenciales complejas relacionadas armónicamente

$$\phi_k(t) = e^{jk\omega_0 t}, \quad k=0, \pm 1, \pm 2, \dots$$

Cada una de estas señales tiene una frecuencia fundamental que es múltiplo de ω_0 y por lo tanto cada una es periódica con periodo T_0 (aunque para $|k| \geq 2$ el periodo fundamental de $\phi_k(\tau)$ es una fracción de T_0). Entonces, una combinación lineal de exponenciales complejas relacionadas armónicamente de la siguiente forma:

$$x(t) = \sum_{k=-\infty}^{\infty} a_k e^{jk\omega_0 t} \quad (2-3)$$

es también periódica con periodo T_0 . En la ecuación (2-3) el término para $k=0$ es un término constante o de CD, los dos términos para $k=+1$ y $k=-1$ tienen periodo fundamental igual a T_0 y se conocen en conjunto como los componentes fundamentales o componentes de la primera armónica. Los términos para $k=+2$ y $k=-2$ son periódicos con la mitad del periodo (o equivalente, con el doble de la frecuencia) de los componentes fundamentales y se conocen como los componentes de la segunda armónica. De manera general, los componentes para $k=+N$ y $k=-N$ son los componentes de la N ésima armónica.

La representación de una señal periódica de la forma de la ecuación (2-3) se conoce como la representación en serie de Fourier.

Los coeficientes a_k de ésta ecuación se calculan con la ecuación

$$a_k = \frac{1}{T_0} \int_0^{T_0} x(t) e^{-jk\omega_0 t} dt \quad (2-4)$$

Las ecuaciones (2-3) y (2-4) definen la serie de Fourier, la ecuación (2-3) se refiere con frecuencia como la ecuación de síntesis, mientras que la ecuación (2-4) como la ecuación de análisis.

Un ejemplo, es la descomposición en frecuencia de una onda cuadrada, como muestra la figura 2.5, en la cual se puede ver cómo a medida que calculamos mas componentes armónicos, la señal que recuperemos se asemejará mejor a la señal original.

Observando la figura 2.5, podemos apreciar que ha medida que se aumenta el cálculo de componentes armónicos se presenta un fenómeno en particular, el cual consiste en la presencia de rizos de alta frecuencia y sobrepasos cerca de las discontinuidades, a este fenómeno se le conoce como el fenómeno de Gibbs, en honor al físico matemático Josiah Gibbs, quien investigó y reportó su explicación en 1899.

Estos rizos se pueden comprimir hacia las discontinuidades haciendo mayor el número de componentes armónicos, pero su amplitud pico permanece constante y su amplitud es en promedio cero, es decir que la señal no se modifica, solo que el fenómeno nunca desaparece.

Hasta ahora hemos visto que una señal periódica se puede representar como una combinación lineal de exponenciales complejas relacionadas armónicamente. De hecho, estos resultados se pueden extender para desarrollar una representación de señales no periódicas como una combinación lineal de exponenciales complejas⁹; lo cual es una de las herramientas de Fourier de mayor contribución.

El uso de cualquier transformada, se justifica con el hecho de analizar el comportamiento de las señales en algún sistema o como reaccionará el sistema a una señal determinada; el conocer el dominio de la frecuencia de la señal aporta información difícil de obtener en el dominio del tiempo. Otra ventaja de tener la información de la señal en el dominio de la frecuencia es que los dispositivos y señales reales presentan un ancho de banda limitado, definiendo este último como el intervalo de frecuencia

⁹ V. Oppenheim, Alan V., Willsky, Alan, V.; SEÑALES Y SISTEMAS, Prentice hall, 1994, pp. 176

en el cual la magnitud de la señal es mayor que $1/\sqrt{2}$ multiplicado (dentro de 3db) por su valor en la mitad del intervalo, esto es cierto cuando la respuesta al sistema es constante. El ancho de banda de un dispositivo o de una señal, cuya representación se muestra en la figura 2.6 es, $\omega_2-\omega_1$.

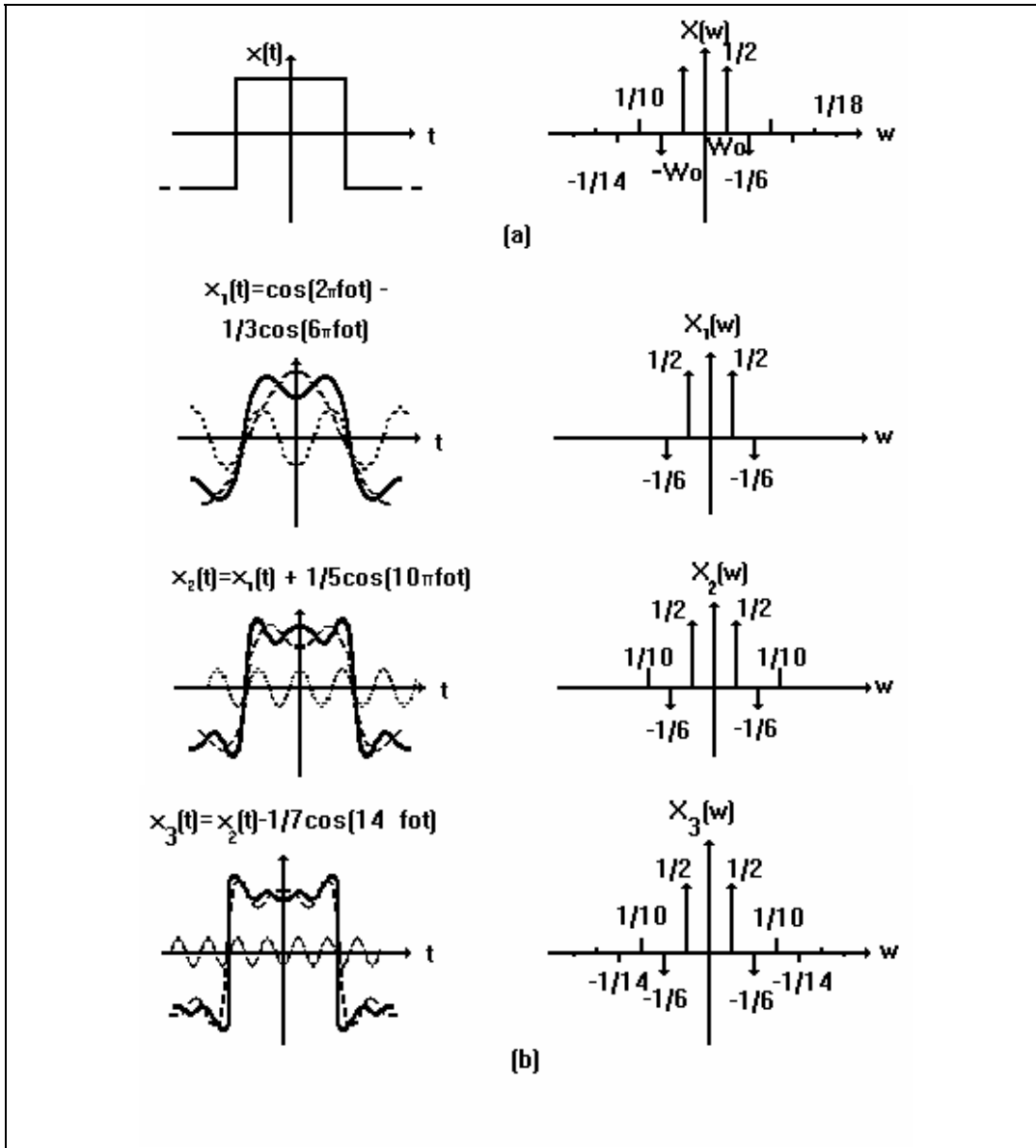


Figura 2.5 Representación de una onda cuadrada en series de Fourier. (a) Función original, (b) Descomposición de la función original en un conjunto de ondas senoidales determinadas por series de Fourier.

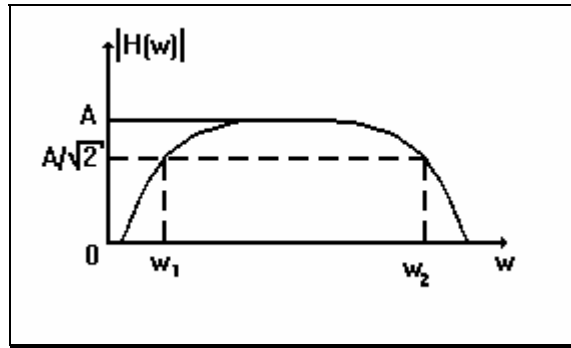


Figura 2.6 Representación gráfica de ancho de banda.

Con lo anterior se determina la importancia de conocer las características de ancho de banda de señales y dispositivos reales, obteniendo con ello más información de su comportamiento en ambos dominios.

2.2.2.

Transformada de Fourier.

El análisis de frecuencia de señales o sistemas reales, en la mayoría de los casos se apoya en la transformada de Fourier, la cual es una extensión de las series de Fourier.¹⁰

La transformada de Fourier de tiempo continuo se define como

$$x(t) = \frac{1}{2\pi} \int_{-\infty}^{\infty} X(\omega) e^{j\omega t} d\omega$$

$$X(\omega) = \int_{-\infty}^{\infty} x(t) e^{-j\omega t} dt$$

donde la segunda ecuación se conoce como la transformada de Fourier o la integral de Fourier y la primera ecuación como la transformada inversa de Fourier, para tiempo continuo.

Con el creciente uso de herramientas de cálculo digitales, como es el caso del Analizador Dinámico de Señales, existe también la transformada de Fourier para tiempo discreto. La representación de las señales en el tiempo discretos también se pueden expresar en forma de exponenciales complejas. De igual forma, también existe la serie de Fourier para tiempo discreto, y se define como

¹⁰ V. Oppenheim, Alan V., Willsky, Alan, V., SEÑALES Y SISTEMAS, Prentice Hall, 1994, pp.203

$$x[n] = \sum_{k=0}^{N-1} a_k e^{jk(2\pi/N)n} \quad (2-5)$$

$$a_k = \frac{1}{N} \sum_{n=0}^{N-1} x[n] e^{-jk(2\pi/N)n}$$

Ambas ecuaciones se conocen como el par de la serie de Fourier de tiempo discreto y se denominan de la misma forma que en el tiempo continuo.

Los coeficientes de la serie de Fourier a_k a menudo se denominan como los coeficientes espectrales de $x[n]$. Estos coeficientes especifican una descomposición de $x[n]$ en una suma de N exponenciales complejas relacionadas armónicamente.

La transformada de Fourier de tiempo discreto se define por

$$x[n] = \frac{1}{2\pi} \int_0^{2\pi} X(\Omega) e^{j\Omega n} d\Omega$$

$$X(\Omega) = \sum_{n=-\infty}^{\infty} x[n] e^{-j\Omega n}$$

La primera ecuación se conoce como la transformada de Fourier de tiempo discreto y en ambas ecuaciones como el par de transformada de Fourier.

2.2.3.

Transformada discreta de Fourier (DFT).

El incremento de métodos de tiempo discreto para el análisis y síntesis de señales y sistemas dió como resultado el desarrollo de herramientas muy eficientes para realizar el análisis de Fourier de secuencias de tiempo discreto. Con ello la existencia de una técnica adecuada para utilizarse en una computadora digital o para su implantación en hardware digital, esta técnica es la transformada discreta de Fourier (DFT) para señales de duración finita, se puede desarrollar de la siguiente forma:

Sea $x[n]$ una señal de duración finita, esto es, hay un entero N_1 , tal que

$$x[n] = 0 \text{ fuera del intervalo } 0 \leq n \leq N_1 - 1$$

Se puede construir una señal periódica $\tilde{x}[n]$ que sea igual a $x[n]$ en un periodo. Sea $N \geq N_1$ un entero dado, y sea $\tilde{x}[n]$ periódica con período N tal que

$$\tilde{x}[n] = x[n], \quad 0 \leq n \leq N_1 - 1 \quad (2-6)$$

Los coeficientes de la serie de Fourier para $\tilde{x}[n]$ están dados por

$$a_k = \frac{1}{N} \sum_{n=0}^{N-1} \tilde{x}[n] e^{-jk(2\pi/N)n}$$

seleccionado el intervalo de la sumatoria en el que $\tilde{x}[n] = x[n]$, obtenemos

$$a_k = \frac{1}{N} \sum_{n=0}^{N-1} x[n] e^{-jk(2\pi/N)n}$$

El conjunto de coeficientes definido por la ecuación anterior comprende la DFT de $x[n]$.

La DFT de $x[n]$ usualmente se denota por $\tilde{X}(k)$ y se define como

$$\tilde{X}(k) = a_k = \frac{1}{N} \sum_{n=0}^{N-1} x[n] e^{-jk(2\pi/N)n}, \quad k=0, 1, \dots, N-1 \quad (2-7)$$

La importancia de la DFT radica en varios hechos, primero, podemos observar que la señal original de duración finita puede recuperarse a partir de su DFT. En particular, la ecuación de síntesis (2-5) para la representación en series de Fourier de $\tilde{x}[n]$ nos permite calcular $\tilde{x}[n]$ a partir de a_k . Entonces, usando las ecuaciones (2-6) y (2-7) tenemos:

$$x[n] = \sum_{k=0}^{N-1} \tilde{X}(k) e^{jk(2\pi/N)n}, \quad n = 0, 1, \dots, N-1 \quad (2-8)$$

Por consiguiente, puede considerarse que la señal de duración finita está especificada por un conjunto finito de valores diferentes de cero, o bien por el conjunto finito de valores $\tilde{X}(k)$ en su DFT. Una segunda

característica importante de la DFT es que para su cálculo hay un algoritmo más rápido que se conoce como la transformada rápida de Fourier (FFT).

Por último, la selección de N en la ecuación (2-8) no es fija, siempre que N sea más grande que la duración de $\tilde{x}[n]$. Por esta razón $\tilde{X}(k)$ en la ecuación (2-7) se refiere con frecuencia como la DFT de N puntos para hacer explícita la longitud de la sumatoria en la ecuación (2-7), lográndose un mayor eficiencia computacional a partir de la selección como una potencia de 2.

Podemos ver que para una señal discreta en el tiempo, el espectro es continuo en la frecuencia. La figura 2.7 muestra de forma gráfica las herramientas de Fourier analizadas anteriormente, la figura 2.7(a) muestra a la señal en el dominio del tiempo, que consta de una onda cuadrada; la figura 2.7(b) muestra los componentes armónicos de la señal por medio de series de Fourier; la figura 2.7(c) muestra el espectro de la señal en el dominio de la frecuencia por la transformada de Fourier y por último la figura 2.7(d) muestra a la señal en el dominio de la frecuencia por la DFT.

Como se puede apreciar en la figura 2.7, el cálculo de la DFT proporciona las muestras de la transformada de Fourier.

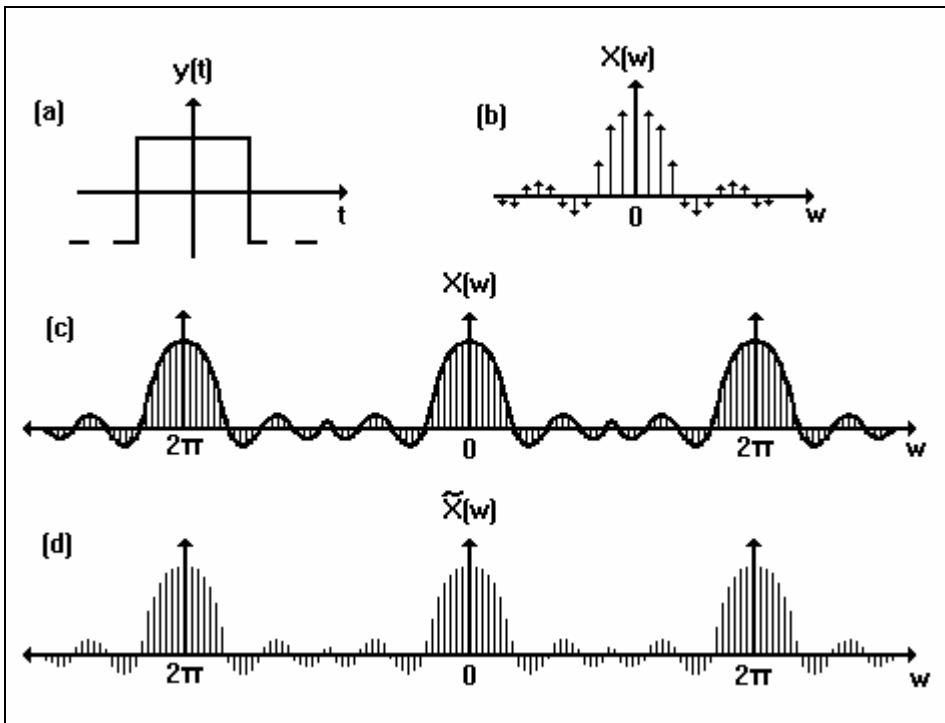


Figura 2.7. Resultados de utilizar las herramientas de Fourier a una señal cuadrada (a). Series de Fourier (b), transformada de Fourier (c) y DFT (d).

Como se puede apreciar en la figura 2.7, el cálculo de la DFT proporciona las muestras de la transformada de Fourier.

2.2.4. Transformada rápida de Fourier.

La transformada rápida de Fourier (FFT, Fast Fourier Transform) es un algoritmo de computadora para el cálculo de la transformada discreta de Fourier (DFT), de hecho se conoce como FFT, cualquier algoritmo que realice menos operaciones de multiplicación y de suma que aplicando la DFT directamente.

La FFT de diezmado en el tiempo funciona partiendo la sucesión completa $\{\tilde{X}_k\}$ de la ecuación (2-7) en un cierto número de sucesiones más cortas. No se calcula la DFT de la sucesión original, sino las de estas secuencias más cortas. A continuación la FFT combina estas DFT de forma ingeniosa para dar la DFT completa de $\{\tilde{X}_k\}$. Esto suena complicado, pero en realidad el fundamento lógico es sorprendentemente sencillo.

Supongamos que $\{\tilde{X}_k\}$, $k=0,1,\dots,N-1$, es la sucesión de la figura 2.8(a), donde N es un número par, y que la partimos en dos sucesiones más cortas, $\{Y_k\}$ y $\{Z_k\}$, como se muestra en la figura 2.8(b), siendo:

$$\left. \begin{array}{l} Y_k = \tilde{X}_{2k} \\ Z_k = \tilde{X}_{2k+1} \end{array} \right\} k = 0, 1, 2, \dots, (N/2) - 1$$

La DFT de estas dos sucesiones cortas son, según (2-7),

$$Y_n = \frac{1}{(N/2)} \sum_{k=0}^{N/2-1} y_k e^{-j\frac{2\pi nk}{(N/2)}} \quad n = 0, 1, 2, \dots, (N/2) - 1 \quad (2-10)$$

$$Z_n = \frac{1}{(N/2)} \sum_{k=0}^{N/2-1} z_k e^{-j\frac{2\pi nk}{(N/2)}}$$

Volvamos ahora a la DFT de la sucesión original $\{\tilde{X}_k\}$ y reordenemos la suma en dos sumas separadas similares a las que aparecen en (2-10). En primer lugar separamos los términos pares e impares de $\{\tilde{X}_k\}$:

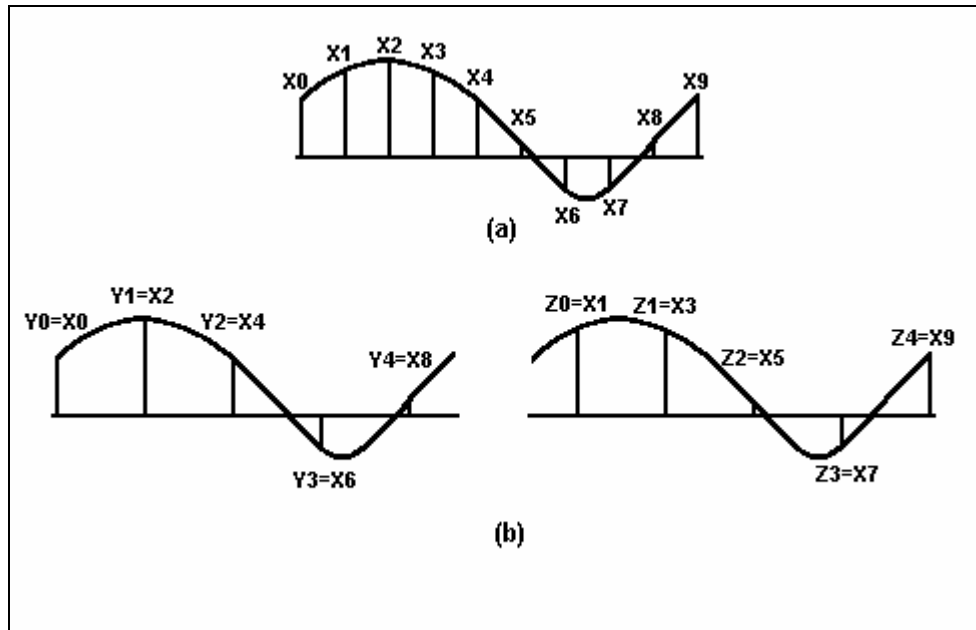


Figura 2.8. Partición de (a) la sucesión original $\{\tilde{X}_k\}$ en (b) dos semisucesiones $\{Y_k\}$ y $\{Z_k\}$

$$\begin{aligned}
 X_n &= \frac{1}{N} \sum_{k=0}^{N-1} x_k e^{-j\frac{2\pi kn}{N}} \\
 &= \frac{1}{N} \left\{ \sum_{k=0}^{N/2-1} x_{2k} e^{-j\frac{2\pi(2k)n}{N}} + \sum_{k=0}^{N/2-1} x_{2k+1} e^{-j\frac{2\pi(2k+1)n}{N}} \right\}
 \end{aligned}$$

Entonces, sustituyendo (2-9) en esta expresión,

$$X_n = \frac{1}{N} \left\{ \sum_{k=0}^{N/2-1} y_k e^{-j\frac{2\pi kn}{N/2}} + e^{-j\frac{2\pi n}{N}} \sum_{k=0}^{N/2-1} z_k e^{-j\frac{2\pi kn}{N/2}} \right\}$$

y comparando con (2-10) encontramos que

$$X_n = \frac{1}{2} \{ Y_n + e^{-j(2\pi n/N)} Z_n \}, \quad n = 0, 1, 2, \dots, (N/2) - 1 \quad (2-11)$$

Así, pues, la DFT de la sucesión original puede obtenerse directamente a partir de las DFT de las dos semisucesiones Y_n y Z_n según la ecuación (2-11). Esta ecuación es la base del método FFT de diezmo en el tiempo. Si el número de muestras N de la sucesión $\{\tilde{X}_k\}$ es una potencia de 2, las semisucesiones $\{Y_k\}$ y $\{Z_k\}$ pueden a su vez ser partidas en semi-

semisecuencias, y así sucesivamente, hasta que finalmente las últimas subsucesiones contengan cada una dos términos.

La fórmula (2-11) es válida solamente para valores de n comprendidos entre 0 y $N/2-1$, es decir, solamente se aplica a la mitad de los coeficientes de la serie $\{X_n\}$. Pero nosotros necesitamos los valores de X_n para todo n comprendido entre 0 y $(N-1)$. La expresión (2-11) es, pues, solamente la mitad; para obtener la otra mitad, correspondiente a $N/2 \leq n \leq (N-1)$, tendremos en cuenta el hecho de que, como hemos visto, Y_n y Z_n son periódicas en n y se repiten con periodo $N/2$, de forma que

$$\begin{aligned} Y_{n-N/2} &= Y_n \\ Z_{n-N/2} &= Z_n \end{aligned}$$

La formulación completa para el cálculo de los valores de X_n a partir de Y_n y Z_n es

$$\begin{aligned} X_n &= \frac{1}{2} \left\{ Y_n + e^{-j(2\pi n/N)} Z_n \right\}, \quad n=0,1,2,\dots,(N/2-1) \\ X_n &= \frac{1}{2} \left\{ Y_{n-N/2} + e^{-j(2\pi n/N)} Z_{n-N/2} \right\}, \quad n=N/2,(N/2+1),\dots,(N-1) \end{aligned}$$

o bien, si queremos que n varíe solamente entre 0 y $N/2$, una forma alternativa es

$$\begin{aligned} X_n &= \frac{1}{2} \left\{ Y_n + e^{-j(2\pi n/N)} Z_n \right\} \\ & \quad , \quad n=0,1,2,\dots,(N/2)-1 \\ X_{n+N/2} &= \frac{1}{2} \left\{ Y_n + e^{-j(2\pi(n+N/2)/N)} Z_n \right\} \end{aligned}$$

Recordando que $e^{-j\pi} = -1$, esta expresión se puede simplificar a

$$\begin{aligned} X_n &= \frac{1}{2} \left\{ Y_n + e^{-j(2\pi n/N)} Z_n \right\} \\ X_{n+N/2} &= \frac{1}{2} \left\{ Y_n - e^{-j(2\pi n/N)} Z_n \right\} \end{aligned}, \quad n=0,1,2,\dots,(N/2)-1$$

Finalmente, si definimos una nueva variable compleja:

$$W = e^{-j(2\pi/N)} \tag{2-12}$$

podemos obtener la denominada "mariposa" de cálculo

$$X_n = \frac{1}{2} \{Y_n + W^n Z_n\}$$

$$X_{n+N/2} = \frac{1}{2} \{Y_n - W^n Z_n\}$$

que aparece en la mayor parte de los programas de cálculo de FFT.

A continuación se muestra un ejemplo para justificar el algoritmo de la FFT.

Consideremos la ecuación (2-7) de la DFT, la cual describe el cálculo de N multiplicaciones, por ejemplo si $N=4$ tenemos que la ecuación (2-12) se puede escribir como:

$$X(0) = x_0(0)W^0 + x_0(1)W^0 + x_0(2)W^0 + x_0(3)W^0$$

$$X(1) = x_0(0)W^0 + x_0(1)W^1 + x_0(2)W^2 + x_0(3)W^3$$

$$X(2) = x_0(0)W^0 + x_0(1)W^2 + x_0(2)W^4 + x_0(3)W^6$$

$$X(3) = x_0(0)W^0 + x_0(1)W^3 + x_0(2)W^6 + x_0(3)W^9$$

estas ecuaciones se pueden representar fácilmente en forma de matriz

$$\begin{bmatrix} X(0) \\ X(1) \\ X(2) \\ X(3) \end{bmatrix} = \begin{bmatrix} W^0 & W^0 & W^0 & W^0 \\ W^0 & W^1 & W^2 & W^3 \\ W^0 & W^2 & W^4 & W^6 \\ W^0 & W^3 & W^6 & W^9 \end{bmatrix} \begin{bmatrix} x_0(0) \\ x_0(1) \\ x_0(2) \\ x_0(3) \end{bmatrix} \quad (2-13)$$

y de forma más compacta como

$$X(n) = W^{nk} x_0(k)$$

Examinando la matriz (2-13) podemos apreciar que W y posiblemente $x_0(k)$ son complejos, por lo que se requieren N^2 operaciones de multiplicación y $N(N-1)$ operaciones de suma para resolver el cálculo de tal matriz. La importancia de la FFT es que reduce el número de operaciones de multiplicación y suma en el cálculo de la matriz (2-13).

Para ilustrar el algoritmo de la FFT, es conveniente elegir el número de puntos de muestra de $x_0(k)$ de acuerdo a la relación $N = 2^\gamma$, donde γ es un entero. Recordando que la ecuación (2-13) resultó de haber elegido $N=4=2^\gamma=2^2$, ahora podemos proceder al cálculo de la FFT.

El primer paso es escribir la ecuación (2-13) de la siguiente forma:

$$\begin{bmatrix} X(0) \\ X(1) \\ X(2) \\ X(3) \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & W^1 & W^2 & W^3 \\ 1 & W^2 & W^0 & W^2 \\ 1 & W^3 & W^2 & W^1 \end{bmatrix} \begin{bmatrix} x_0(0) \\ x_0(1) \\ x_0(2) \\ x_0(3) \end{bmatrix} \quad (2-14)$$

esta matriz se deriva al utilizar la relación

$$W^{nk} = W^{nk \bmod(N)}$$

donde $[nk \bmod(N)]$ es el residuo de la división de nk por N , por lo que si $N=4$, $n=2$ y $k=3$ tenemos:

$$W^6 = W^2$$

ya que:

$$\begin{aligned} W^{nk} = W^6 &= \exp\left[\left(-\frac{j2\pi}{4}\right)(6)\right] = \exp[-j3\pi] \\ &= \exp[-j\pi] = \exp\left[\left(-\frac{j2\pi}{4}\right)(2)\right] = W^2 = W^{nk \bmod(N)} \end{aligned}$$

el siguiente paso es factorizar la matriz cuadrada (2-14) como:

$$\begin{bmatrix} X(0) \\ X(2) \\ X(1) \\ X(3) \end{bmatrix} = \begin{bmatrix} 1 & W^0 & 0 & 0 \\ 1 & W^2 & 0 & 0 \\ 0 & 0 & 1 & W^1 \\ 0 & 0 & 1 & W^3 \end{bmatrix} \begin{bmatrix} 1 & 0 & W^0 & 0 \\ 0 & 1 & 0 & W^0 \\ 1 & 0 & W^2 & 0 \\ 0 & 1 & 0 & W^2 \end{bmatrix} \begin{bmatrix} x_0(0) \\ x_0(1) \\ x_0(2) \\ x_0(3) \end{bmatrix} \quad (2-15)$$

Como se puede ver, se intercambiaron las filas 1 y 2 (las filas se han numerado como 0, 1, 2 y 3). Se han intercambiado para reescribir la columna del vector $X(n)$, permitiendo el intercambio del vector a :

$$\bar{X}(n) = \begin{bmatrix} X(0) \\ X(2) \\ X(1) \\ X(3) \end{bmatrix}$$

Con el cambio anterior y la factorización, el algoritmo de la FFT logra una mayor eficiencia.

Analizando la matriz (2-15), podemos examinar el número de multiplicaciones que se requieren para resolver esta ecuación, por lo que

$$\begin{bmatrix} x_1(0) \\ x_1(1) \\ x_1(2) \\ x_1(3) \end{bmatrix} = \begin{bmatrix} 1 & 0 & W^0 & 0 \\ 0 & 1 & 0 & W^0 \\ 1 & 0 & W^2 & 0 \\ 0 & 1 & 0 & W^2 \end{bmatrix} \begin{bmatrix} x_0(0) \\ x_0(1) \\ x_0(2) \\ x_0(3) \end{bmatrix}$$

en la cual, el vector $x_1(k)$ es igual al producto de las dos matrices de la derecha de la ecuación (2-15).

El elemento $x_1(0)$ se calcula con una multiplicación compleja y una suma compleja (W^0 no se reduce a la unidad para desarrollar un resultado generalizado).

$$x_1(0) = x_0(0) + W^0 x_0(2) \quad (2-16)$$

El elemento $x_1(1)$ también se determina al realizar una operación de multiplicación y otra de suma, ambas complejas. En el caso de $x_1(2)$ solo se requiere una operación de suma compleja, debido a que $W^0 = -W^2$; por lo que:

$$\begin{aligned} x_1(2) &= x_0(0) + W^2 x_0(2) \\ &= x_0(0) - W^0 x_0(2) \end{aligned}$$

donde las multiplicaciones complejas $W^0 x_0(2)$ se originan al determinar $x_1(0)$ en la ecuación (2-16). Por la misma razón, $x_1(3)$ no requiere de operaciones de multiplicación. El vector intermedio $x_1(k)$ se determina por cuatro operaciones de suma y dos de multiplicación.

Continuando con el cálculo de (2-15)

$$\begin{bmatrix} X(0) \\ X(2) \\ X(1) \\ X(3) \end{bmatrix} = \begin{bmatrix} x_2(0) \\ x_2(1) \\ x_2(2) \\ x_2(3) \end{bmatrix} = \begin{bmatrix} 1 & W^0 & 0 & 0 \\ 1 & W^2 & 0 & 0 \\ 0 & 0 & 1 & W^1 \\ 0 & 0 & 1 & W^3 \end{bmatrix} \begin{bmatrix} x_1(0) \\ x_1(1) \\ x_1(2) \\ x_1(3) \end{bmatrix}$$

El término $x_2(0)$ se determina por una multiplicación y una suma compleja.

$$x_2(0) = x_1(0) + W^0 x_1(1)$$

El elemento $x_2(1)$ se calcula por una suma ya que $W^0 = -W^2$. De manera similar, $x_2(2)$ se determina por una suma y una multiplicación. Por último $x_2(3)$ se determina solo por una suma.

El cálculo de $\bar{X}(n)$ definida en (2-15) requiere de un total de cuatro multiplicaciones complejas y ocho sumas complejas. El cálculo de $X(n)$ en (2-13) requiere de 16 multiplicaciones complejas y 12 sumas complejas. Si observamos, el proceso de factorización de la matriz introduce ceros en la matriz factorizada y, como resultado, reduce el número de multiplicaciones requerido a una potencia de 2. Si tomamos en cuenta el tiempo de cálculo que se requiere para realizar las multiplicaciones, se justifica la eficiencia de la FFT.

Para $N = 2^\gamma$ el algoritmo de la FFT es simplemente un procedimiento de factorización de una matriz de $N \times N$ en γ matrices (con dimensiones de $N \times N$) tal que cada una de las matrices factorizadas tiene una propiedad especial de reducir el número de multiplicaciones y sumas complejas con relación de $N \log N$, en comparación con el cálculo directo que requiere de $N(2N-1) \approx N^2$. Si extendemos los resultados de este ejemplo, podemos decir que la FFT requiere de $N\gamma/2=4$ multiplicaciones complejas y de $N\gamma=8$ sumas complejas, comparando con el método directo que requiere de N^2 multiplicaciones complejas y $N(N-1)$ sumas complejas. Si asumimos que el tiempo de cálculo es proporcional al número de multiplicaciones, entonces podemos aproximar el tiempo de cálculo de la FFT como:

$$\frac{N^2}{N\gamma/2} = \frac{2N}{\gamma}$$

Por ejemplo, para $N=1024=2^{10}$ el cálculo se reduce a más de 200 a 1. La figura 2-9 muestra la relación entre el número de multiplicaciones requeridas usando el algoritmo de la FFT comparado con el número de multiplicaciones usando el método directo.

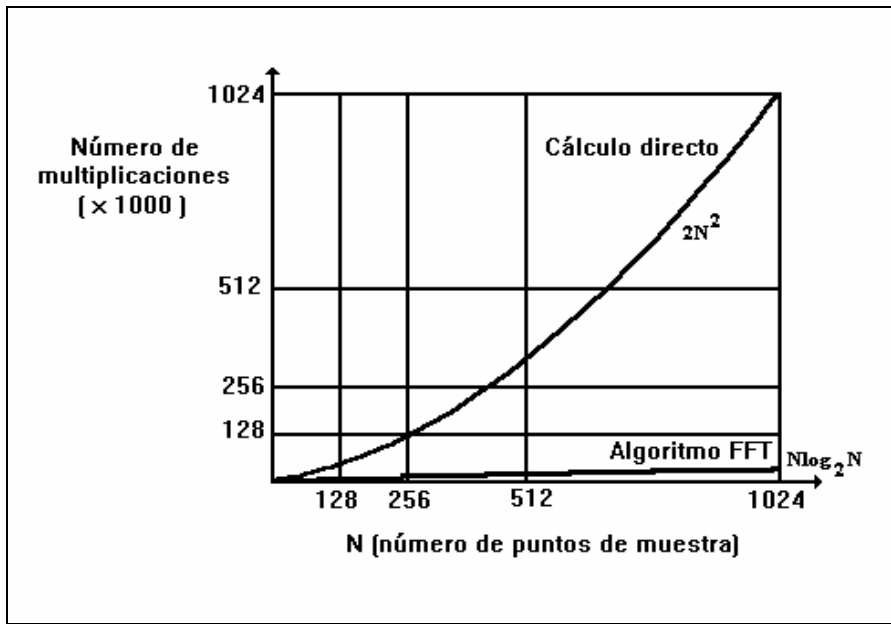


Figura 2.9 Comparación del número de multiplicaciones requeridas por el cálculo directo y el algoritmo de la FFT.

2.3.

Conversión analógica a digital.

Como se mencionó en el capítulo 1, una de las partes fundamentales de una tarjeta de adquisición de señales es el convertidor analógico a digital, ya que es de suma importancia cuando se interactúa con un sistema digital como es la computadora.

La necesidad de procesar información y controlar funciones usando métodos digitales ha venido incrementándose en la actualidad. Sin embargo, como se ha mencionado anteriormente, los datos del mundo real se encuentran en forma analógica. Un sistema de adquisición de datos proporciona la interfaz necesaria para pasar de un tipo de datos a otro, la cual se conoce como interfaz Analógica a Digital o A/D, y se encarga de convertir los datos originales de uno o más sensores o transductores en una señal compatible al procesamiento digital usando componentes tales como amplificadores, filtros, circuitos de muestreo y retención, multiplexores y convertidores A/D.

El convertidor A/D, es el componente más importante en cualquier sistema de adquisición de datos, a continuación analizaremos los fundamentos, las técnicas de selección y la interfaz de los convertidores A/D.

2.3.1.

Relación Entrada / Salida.

La conversión A/D es esencialmente una operación de proporciones. La señal analógica de entrada v_i se convierte en una fracción x para compararla contra una señal de referencia V_r . La salida digital de un convertidor es una representación codificada de esta fracción. La figura 2.10(a) muestra esta relación fundamental. Si el código de salida del convertidor consta de n bits, el número de niveles discretos de salida se fija a 2^n . Para una correspondencia de uno a uno, el rango de entrada debe cuantizarse en este mismo número de niveles. En el que cada nivel (Quantum) es el valor analógico por el cual dos códigos adyacentes difieren. A esto se le conoce como la medida del bit menos significativo (LSB). Estos es:

$$Q = \text{LSB} = \frac{FS}{2^n}$$

Donde Q es un quantum, LSB se refiere al valor analógico de un LSB y FS es la escala mayor del nivel de entrada analógico.

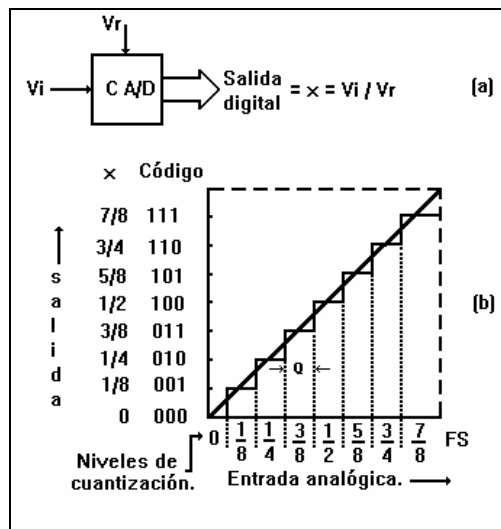


Figura 2.10. Conversión A/D (a) Diagrama de bloques y (b) Relación Entrada/Salida.

Todos los valores analógicos con un quantum dado se representan por el mismo código digital, el cual generalmente corresponde al valor del rango medio llamado umbral (threshold).

Sin embargo existe una incertidumbre en la cuantización de $\pm 1/2$ LSB en cualquier proceso de conversión A/D. Este efecto solo se puede minimizar al incrementar el número de bits en el código de salida del convertidor. La figura 2.10(b) muestra esta relación de conversión para un convertidor A/D de tres bits. La medida LSB es $1/8$ FS y el rango de entrada se cuantiza en ocho niveles distintos de 0 a $7/8$ FS. Se debe tener en cuenta que la salida máxima que es el número binario 111 no corresponde a la escala completa pero si a $7/8$ FS. Con uno de los códigos asignados al nivel cero, la salida máxima de un convertidor A/D siempre corresponderá a un valor analógico de la escala completa menos 1 LSB.

2.3.2.

Errores de conversión.

En Teoría se puede diseñar un convertidor A/D ideal, pero ya en la práctica su funcionalidad puede variar de sus características ideales, las cuales se muestran en la figura 2.10(b). La función de transferencia puede variar de diferentes formas de acuerdo a sus valores ideales. A esto se le conoce como error "offset" o de cero [figura 2.11(a)], y se define como el valor analógico que difiere de la función de transferencia al pasar por cero. La inclinación de la función de transferencia también puede diferir de su valor ideal, generando un "error de inclinación" o "error de ganancia" [figura 2.11(b)]. La mayoría de convertidores A/D que existen en el mercado presentan este tipo de errores, pero se pueden eliminar virtualmente al realizar ajustes iniciales. Existe, sin embargo, otro tipo de errores que son más difíciles de eliminar y que no se pueden modificar o alterar por medio de ajustes, a estos se les conoce como "errores de linealidad" y pueden ser de dos tipos; el primero se conoce como "error de linealidad integral" y se define como la máxima desviación que sufre la función de transferencia con respecto a su línea recta de características ideales cuando los errores de offset y de ganancia son cero [figura 2.11(c)]. El otro tipo de error de linealidad es el "error de linealidad diferencial" que es la desviación que sufre cualquier quantum de su valor analógico ideal, es necesario hacer notar que cuando este error excede 1 LSB, la salida puede tener un código perdido, como se muestra en la figura 2.11(d).

2.3.3.

Resolución del convertidor.

Este parámetro es muy importante al definir un convertidor, por que define el menor cambio requerido en la entrada analógica al convertidor A/D, para cambiar su código de salida por un nivel de 1. Generalmente se especifica asumiendo un convertidor ideal y entonces refleja la capacidad del convertidor en su ejecución actual. Se puede dar en función de la escala mayor, en milivolts para un rango de entrada dado o simplemente como el número de bits de conversión, éste último es el término comúnmente usado.

2.3.4.

Precisión del convertidor.

Para los convertidores A/D la precisión se define como la diferencia entre el voltaje de entrada actual y su equivalente de más alta escala en el código binario de salida. Se conoce como precisión absoluta cuando se especifica en volts. Comúnmente se especifica relacionándolo con la señal analógica en medida del LSB y es cuando toma el nombre de precisión relativa. En cualquier caso, es la suma total de todos los errores de conversión, incluyendo el error de cuantización. Las especificaciones de errores del convertidor generalmente se dan en términos de LSB. Para convertidores que no requieren de un previo ajuste de offset o de ganancia, tal como el convertidor A/D ADC0809 de National Semiconductor, cuyo error de ajuste total se puede especificar como $\pm 1/2$ LSB.

2.3.5.

Tiempo y velocidad de conversión.

Después de que el convertidor A/D recibe el comando de inicio, requiere de un tiempo, llamado tiempo de conversión t_c , para que el convertidor pueda proporcionar datos de salida válidos. El voltaje de entrada cambia durante el proceso de conversión, lo cual introduce una salida que refleja incertidumbre y que por lo tanto es indeseable. La precisión de la conversión máxima se realiza solamente si esta incertidumbre se mantiene por debajo de la resolución del convertidor. Por ejemplo, para un convertidor de n-bits, tiene un tiempo de conversión t_c .

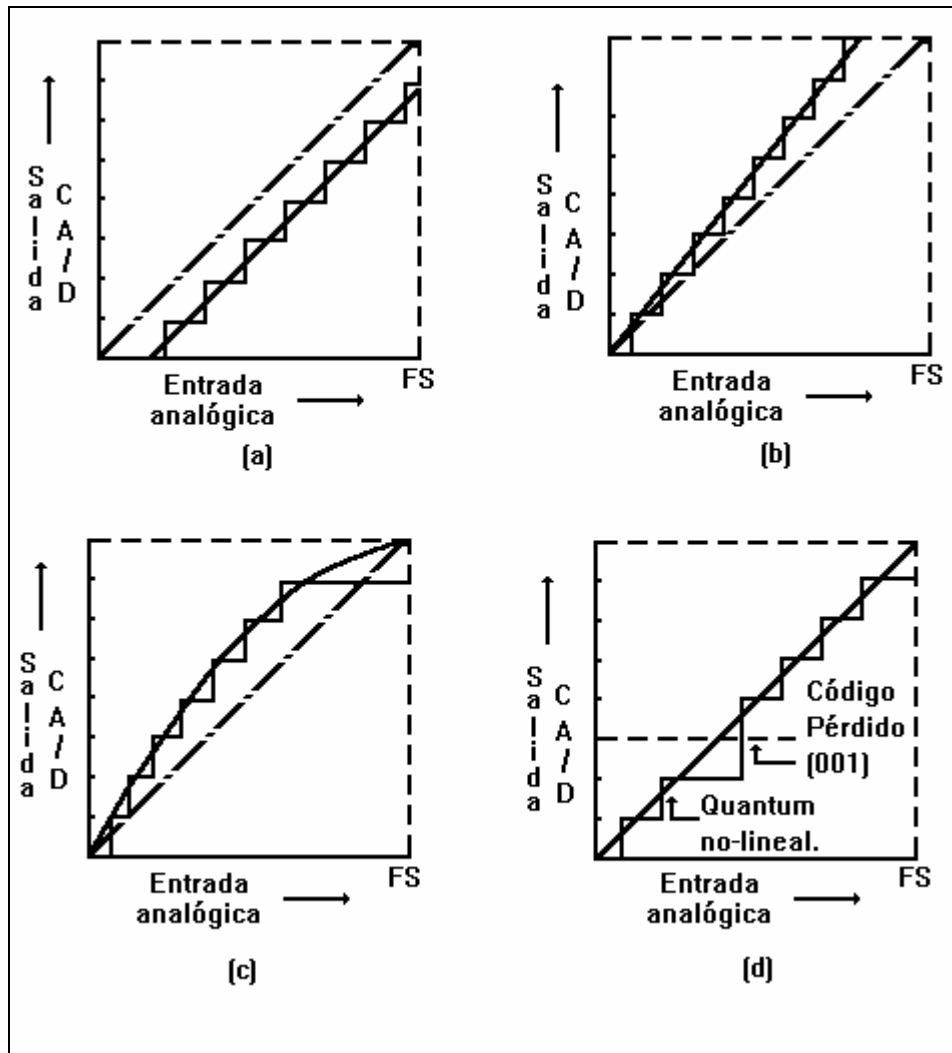


Figura 2.11. Errores de conversión. (a) Offset, (b) De ganancia, (c) De linealidad integral y (d) De linealidad diferencial.

$$\left(\frac{dV}{dt}\right)_{max} \leq \frac{FS}{2^n t_c}$$

Por ejemplo, se considera como entrada una señal senoidal con amplitud A y frecuencia f para un convertidor monolítico de 8-bits, con un tiempo de conversión de $100 \mu s$:

$$V_i = A \sin(2\pi ft)$$

La razón de cambio de la señal de entrada es dada por

$$\left(\frac{dV}{dt}\right) = 2\pi f A \cos(2\pi ft)$$

y la máxima razón de cambio :

$$\left(\frac{dV_i}{dt}\right)_{max} = 2\Pi f A$$

Si igualamos la escala máxima FS a 2A, el valor Pico a Pico de la onda senoidal será:

$$2\Pi f A \leq \frac{2A}{2^n t_c}$$

$$f \leq \frac{1}{2^n t_c \Pi}$$

$$f_{max} = \frac{1}{2^n t_c \Pi} = 12.4\text{Hz}$$

Esta frecuencia es muy baja para la mayoría de aplicaciones, para solucionar el problema, se utilizan circuitos de muestreo y retención (Sample and Hold S/H) entre la señal de entrada y el convertidor. Un circuito de S/H es un simple circuito análogo que muestrea de forma rápida la señal de entrada y la mantiene hasta que el convertidor finaliza la conversión. El intervalo de tiempo que se necesita es ahora un retardo de tiempo llamado tiempo de apertura, t_a . Este retardo se refiere al tiempo durante el cual el circuito de S/H almacena y retiene la información de la señal y comúnmente es del orden de pocos nanosegundos. Si se usa un circuito de S/H con una tiempo de apertura de 20ns se calcula nuevamente la máxima frecuencia permitida,

$$f_{max} = \frac{1}{2^8 t_a \pi} = 62.17\text{kHz}$$

la cual es una frecuencia razonable para un convertidor de 100 μ s. La frecuencia máxima se puede manipular por medio del tiempo de apertura del circuito de S/H.

En este caso, el circuito ADC0809 cuenta con un circuito de S/H internamente, evitando el uso de un circuito de S/H exterior.

2.4.

Convertidor analógico a digital por aproximaciones sucesivas.

La figura 2.12 muestra el diagrama a bloques de un registro (ADC) por aproximaciones sucesivas. Consta de un convertidor digital a analógico, un comparador y un registro de aproximación sucesiva (SAR). Se necesita una terminal para el voltaje de entrada analógica V_{ent} . La salida digital está disponible; en forma serie o paralela. Se requiere un mínimo de tres terminales de control: *inicia la conversión*, inicia la secuencia de conversión A/D, *fin de conversión* indica cuándo se termina la conversión y una terminal externa de reloj para establecer el tiempo de conversión.

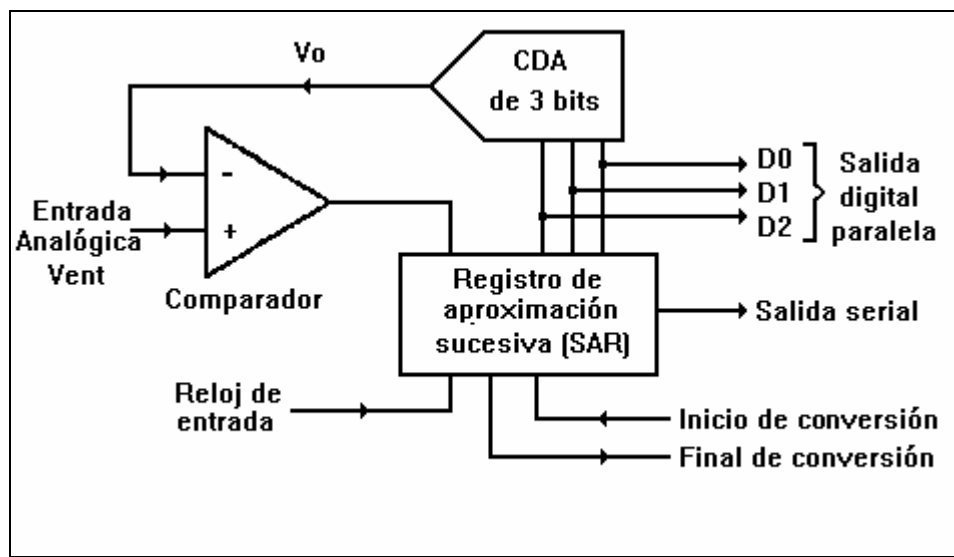


Figura 2.12 Diagrama de bloques de un Convertidor Analógico a Digital por aproximaciones sucesivas de 3 bits.

2.4.1.

Operación del circuito.

En base a la figura 2.12, la orden de *inicia conversión*, inicia el ciclo de conversión analógica a digital. El registro de aproximación sucesiva (SAR) conecta la secuencia de números digitales, un número por cada bit, a las entradas del convertidor digital a analógico.

El convertidor digital a analógico transforma cada número digital en una salida analógica V_o . El voltaje analógico de entrada se compara con V_o . El comparador le dice al SAR cuando V_{ent} es mayor o menor que la salida

del convertidor Digital a analógico, V_o . Para cada bit de la salida de 3 bits, deben efectuarse tres comparaciones.

Las comparaciones se hacen comenzando con el bit más significativo y terminan con el bit menos significativo, como se explicará a continuación. Al terminar la comparación el registro de aproximación sucesiva (SAR) envía la señal que finalizó la conversión. El equivalente digital de V_{ent} está ahora presente en la salida digital del registro.

2.4.2.

Analogía por aproximaciones sucesivas.

Suponga que se tienen los pesos de 1, 2 y 4 libras (SAR) y una balanza (comparador y convertidor digital a analógico). Considere un peso de 1 lb como 1 LSB y el peso más significativo de 4 lb como 4 LSB. Ver las figuras 2.12 y 2.13 para referencia. V_{ent} corresponde a un peso desconocido.

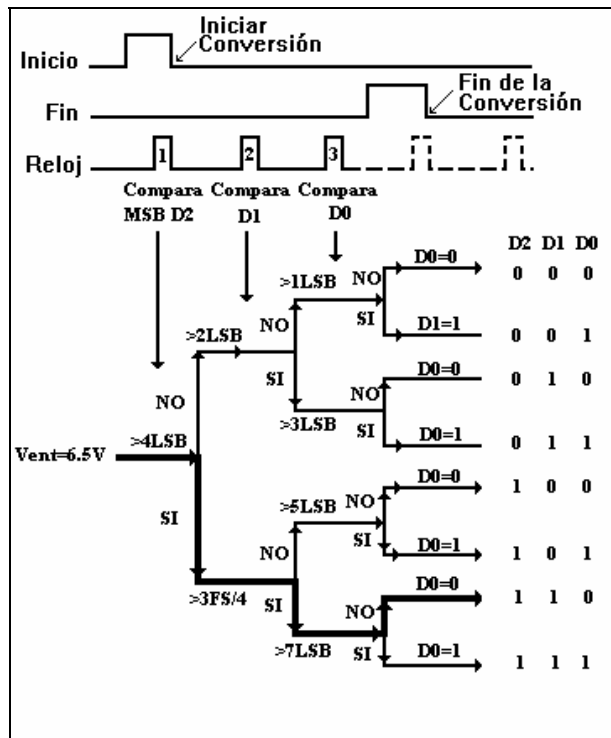


Figura 2.13. Analogía por aproximaciones sucesivas.

Convertimos $V_{ent} = 6.5V$ en una salida digital (peso desconocido = 65 lb). Se pondrá el peso desconocido en un platillo de balanza y el de 4 lb en el otro para comparar si el peso desconocido (V_{ent}) excede 4 lb. El registro

de aproximación sucesiva utiliza un pulso de reloj para aplicar 100 bits (MSB) al convertidor digital a analógico en la figura 2.13. Su salida, $V_o=4\text{ V}$, se compara con V_{ent} . El bit más significativo se hace 1 si $V_{ent} > V_o$. Esto es como dejar el peso de 4 lb en la balanza.

El registro de aproximaciones sucesivas aplica después 110 (se agrega un peso de 2 lb) al convertidor digital a analógico; D_1 se pone a 1 puesto que $V_{ent} = 6.5\text{ V}$ es mayor que $V_o = 6\text{ V}$. Por último, el registro aplica 111 al convertidor digital a analógico (se agrega 1 lb). Dado que $V_{ent} = 6.5\text{ V}$ es menor que 7 V , se pone D_o a cero (se elimina el peso de 1 lb).

2.4.3.

Tiempo de conversión de un convertidor A/D por aproximaciones sucesivas.

La figura 2.13 muestra que se necesita un pulso de reloj para que el registro de aproximación sucesiva (SAR) compare cada bit. No obstante, casi siempre se requiere un pulso adicional para restablecer el registro antes de llevar a cabo la conversión. El tiempo que tarda una conversión analógica a digital dependerá tanto del periodo del reloj T como del número de bits n . La relación es :

$$T_c = T (n + 1)$$

Por ejemplo, un reloj de 1Mhz impulsa un convertidor analógico a digital de aproximaciones sucesivas, para encontrar el tiempo de conversión según la ecuación anterior, el tiempo de un pulso de reloj es de $1\text{ }\mu\text{s}$, por lo que:

$$T_c = 1\text{ }\mu\text{s} (8 + 1) = 9\text{ }\mu\text{s}$$

lo que indica que se necesitan $9\mu\text{s}$ para realizar una conversión.

3

Adquisición y presentación de datos.

La industria de la instrumentación ha sufrido una gran variedad de cambios como resultado de la revolución de las PC's. Estos cambios han ocurrido en ambas áreas, el hardware y el software. La gran mayoría de científicos e ingenieros de todo el mundo usan sus PC's para automatizar sus tareas de investigación, diseño y construcción.

En la actualidad, para obtener una gran ventaja con respecto a la revolución de las PC's, han surgido nuevas herramientas para perfeccionar los conceptos de software y con ello incrementar el potencial de las interfaces de hardware. Estos nuevos conceptos han hecho posible la construcción de sistemas de instrumentación muy poderosos.

La mayoría de sistemas de adquisición de datos se usan para procesar señales analógicas y convertirlas a forma digital para un procesamiento o análisis subsecuente en una computadora o para su transmisión a otros sistemas. La adquisición de datos es la forma de capturar las señales del mundo real y transferirlas a un sistema, como es la computadora; se han desarrollado tarjetas que se insertan en una computadora y realizan la función de interfaz entre las señales y la computadora, este es el caso de la tarjeta que se presenta en el Analizador Dinámico de Señales. La tarjeta de adquisición comúnmente incluye a uno o más convertidores analógico a digital (ADC), convertidores digital a analógico (DAC), puertos digitales de Entrada/Salida o E/S y circuitos contadores o relojes. Estos componentes forman la interfaz entre la computadora y las señales analógicas del mundo real.

Existen muchas ventajas en el uso de tarjetas de adquisición de datos para medición y generación de señales, en comparación con la instrumentación tradicional. Las tarjetas de adquisición de datos cuentan con más flexibilidad y bajos costos, además de que pueden ejecutar una gran variedad de funciones, como las antes mencionadas. Cuentan con grandes rangos de muestreo, su resolución puede variar de 8 a 20 bits; pueden amplificar la señal y se pueden obtener varios canales de entrada, los cuales dependen del diseño de la tarjeta. Estas tarjetas, son apoyadas por herramientas de software, con lo que el potencial se incrementa considerablemente, haciendo más fácil la interfaz con otros sistemas.

En la figura 3.1 se muestra un sistema de adquisición de datos general. Este diagrama muestra los diferentes componentes que se requieren para una interconexión entre las variables físicas y el sistema analizador. En la primera etapa el sistema mide el parámetro físico y lo convierte a una forma eléctrica por medio de un transductor, el cuál debe ser elegido dependiendo del parámetro físico que se desee medir; la señal ya en forma eléctrica alimenta a un amplificador, cuya función es la de convertir (si es necesario) la señal eléctrica a una señal de alto nivel para procesarla en sistemas digitales, los cuales admiten como entrada señales con un rango de 0 a 5 Volts. La salida del transductor puede ser una señal a nivel de milivolts, de alta impedancia, con gran cantidad de ruido o simplemente una señal de corriente. En cualquiera de estos casos el amplificador se usa para convertir dicha señal a un voltaje de alto nivel, la cuál se usa para operar el siguiente circuito analógico. La etapa siguiente al amplificador es un filtro activo paso bajo, para eliminar los componentes de alta frecuencia o ruido de la señal y ejecutar algunas operaciones no lineales sobre la señal, tales como raíz cuadrada, linealización de la señal o multiplicarla por otra función. Con una señal de alto nivel se logra obtener una buena exactitud en la medición.

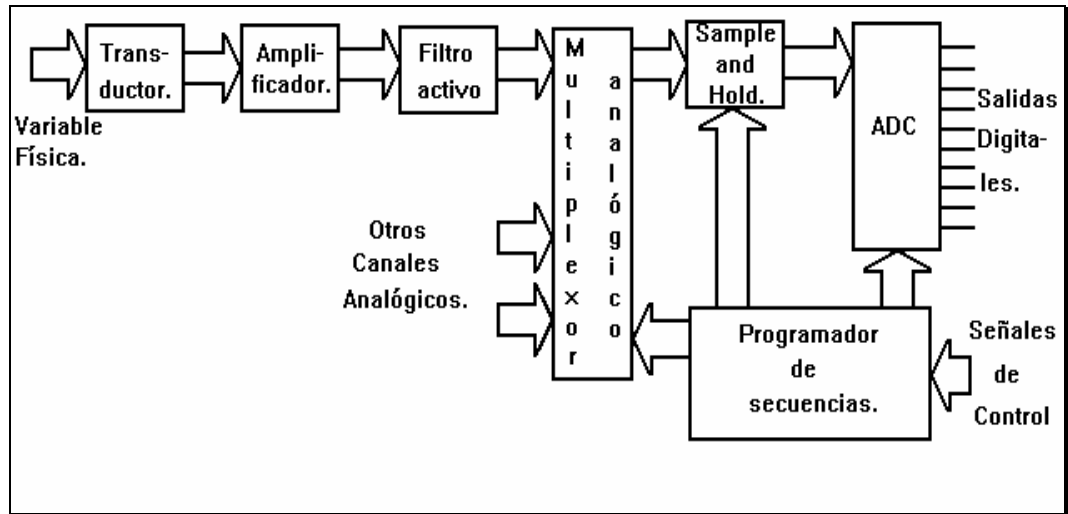


Figura 3.1. Diagrama de un sistema de adquisición de datos general.

La señal pasa a un multiplexor analógico que realiza una operación de conmutación por división en el tiempo entre un número de entradas de señales diferentes (en el caso de contar con varios canales de medición), cada canal se conecta secuencialmente a la salida del multiplexor por un intervalo de tiempo específico. El circuito que se conecta a la salida del multiplexor es el de muestreo y retención (sampling and hold S/H) el cual toma las muestras de la salida del multiplexor a un tiempo especificado y almacena el nivel de voltaje en su salida hasta que el convertidor analógico a digital termina su operación de conversión. El control y el reloj lo realiza un circuito programador de secuencia que a su vez se controla directamente por el microprocesador de la computadora..

En esta parte, se analiza el diseño de la tarjeta de adquisición de datos, así como cada uno de los componentes hardware que se utilizan.

3.1. Acondicionamiento de la señal.

Como se podrá ver en la sección 3.3., el diseño de la tarjeta de adquisición de datos contempla el uso de un convertidor analógico a digital (ADC0809), el cual tiene un rango de voltaje para sus líneas de entrada, de 0 a +5V, para los objetivos del Analizador Dinámico de Señales, es necesario acondicionar la señal de entrada para poder ampliar el rango de voltaje a ± 10 V. Este acondicionamiento se realiza

haciendo un desplazamiento positivo a la señal y realizando una atenuación, se trata de acondicionar la señal sin realizarle modificaciones.

La función de acondicionamiento la realizan los amplificadores operacionales LF353N, en las configuraciones de seguidor y de sumador, éste último cumple con la siguiente ecuación:

$$V_{oc} = - \left[\frac{R_f}{R_d} V_d + \frac{R_f}{R_{ei}} (-V_i) \right]$$

donde:

- V_{co} es el voltaje hacia el convertidor y toma un valor de 0 a 5V.
- V_d es el voltaje para el desplazamiento y su valor es de -12V.
- V_i es el voltaje de la señal y puede tomar un valor de 10 Vpp.
- R_f es la resistencia de lazo cerrado y su valor es de 30 K Ω .
- R_d es la resistencia para el desplazamiento y su valor es de 50K Ω .
- R_{ei} es la resistencia de escala y toma un valor de 150 K Ω .

El primer término (R_f / R_d) V_d proporciona el voltaje de desplazamiento en el acondicionamiento de la señal y es constante igual a 2.5V aproximadamente.

El segundo término (R_f / R_{ei}) V_i es el voltaje de la señal ya modificada, es decir atenuada entre un valor de 0 a +5V.

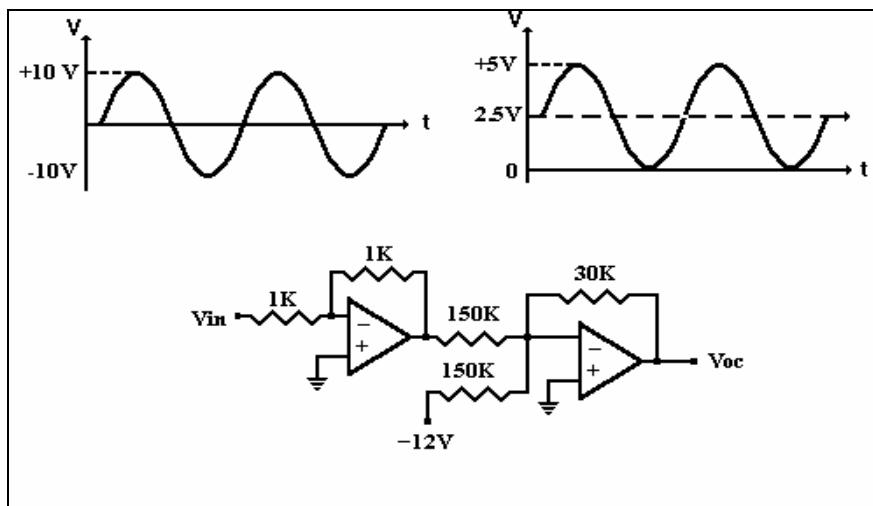


Figura 3.2. Acondicionamiento de la señal.

Antes de emplear la etapa de acondicionamiento de la señal, se utiliza un amplificador operacional en la configuración de amplificador inversor,

el cual se utiliza como protección entre la señal y la tarjeta, además de proporcionar los límites de voltaje permitidos.

El signo negativo de la función se debe a que se utiliza la configuración de sumador inversor, además de que en la primera etapa se hace uso de una configuración de seguidor inversor, con lo que este signo se elimina. La función descrita anteriormente y el circuito se muestran esquemáticamente en la figura 3.2.

3.2.

Diseño de la tarjeta de adquisición de datos.

A continuación se analiza el diseño de la tarjeta de adquisición de datos por módulos y al final se muestra el diagrama completo.

El sistema del Bus de la IBM PC es una extensión del bus del microprocesador Intel 8088, con algunas variantes como son, mayor número de señales para soportar acceso directo a memoria (DMA), interrupciones, técnicas de demultiplexaje y otras. Todas las señales son de nivel TTL (Lógica de transistor a transistor). La figura 3.3 muestra la asignación de las 62 patillas del bus de la IBM PC.

A0 a A19 : Estas 20 patillas se utilizan para direccionar memoria o entrada/salida. A0 es el bit menos significativo (LSB) y A19 es el más significativo (MSB). Estas líneas pueden manejarse directamente por el microprocesador o por el controlador de DMA.

D0 a D7 : Estas 8 patillas son el bus de datos bidireccional. D0 es el menos significativo (LSB) y D7 es el más significativo (MSB). Durante el inicio del ciclo de escritura al bus del microprocesador, éste pone información en el bus de datos antes del corte ascendente al enviar la señal de escritura de entrada/salida (IOW) o escritura a memoria (MEMW), con lo que envía datos a la salida del puerto o de la memoria. Durante un ciclo de lectura del bus, en el bus de datos se pone la información que se encuentra en la entrada del puerto o en la memoria, esto debe ser antes del corte ascendente de la señal de entrada/salida (IOR) o de lectura de memoria (MEMR), con esto se almacenan los datos en el microprocesador.

MEMR, MEMW, IOR, IOW : Estas señales son activo bajo, para controlar las operaciones de lectura o escritura como se mencionó anteriormente.

ALE (address latch enable) : Indica el inicio del ciclo de bus del microprocesador. Cuando se declara la señal, el bus de datos no contiene información.

AEN (address enable) : Esta señal es emitida por el controlador de DMA para indicar que está en proceso un ciclo DMA. Normalmente se usa para deshabilitar la lógica de decodificación del puerto de entrada/salida durante el ciclo de DMA y para que la dirección de memoria no se use como una dirección de puerto de entrada/salida. Esta situación se puede presentar si durante el ciclo de DMA se activan las señales de IOR o IOW.

OSC (oscillator), CLOCK : Esta señal es el sistema de reloj de alta velocidad, por ejemplo, para el microprocesador Intel 8088, esta señal esta disponible con un periodo de 70 ns (14.31818 Mhz), el cual es una tercera parte de la frecuencia del oscilador (4.77 Mhz) del 8088, lo que equivale a un periodo de 210 ns.

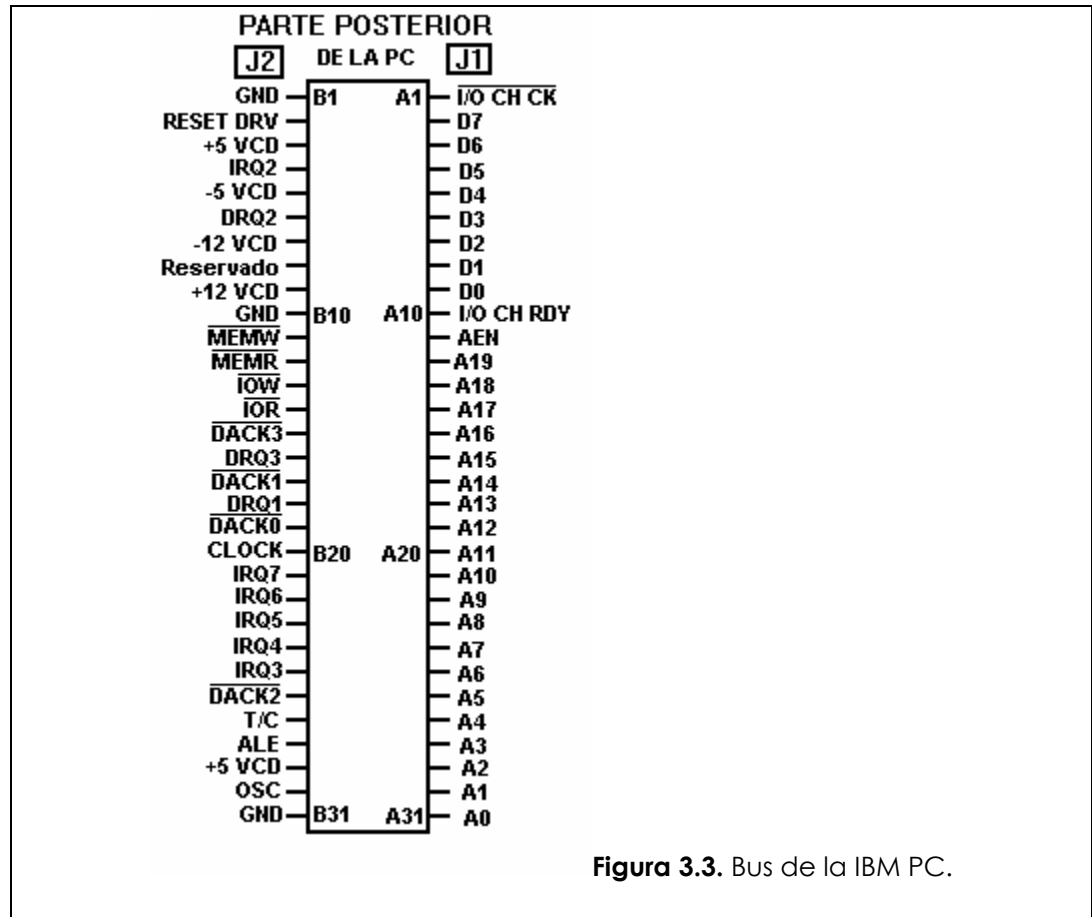
IRQ2 a IRQ7 (interruption request) : Los dispositivo de Entrada/Salida utilizan estas líneas de entrada para generar una petición de interrupción al microprocesador. Estas señales tienen prioridad, IRQ2 tiene la más alta prioridad y IRQ7 tiene la más baja. Una interrupción puede generarse al usar una línea IRQ y mantenerla a un nivel alto hasta que exista un reconocimiento por parte del procesador. Cuando la señal de reconocimiento de interrupción (INTA) no es enviada por el procesador al bus, el reconocimiento normalmente se realiza por un bit del puerto de entrada/salida, enviando el comando OUT a la rutina de servicios de interrupción.

I/O CH RDY (I/O channel ready) : Esta es una señal de entrada usada para generar estados de espera, los cuales aumentan la longitud de los ciclos de bus del microprocesador para memorias o dispositivos de Entrada/Salida lentos.

I/O CH CK (I/O channel check) : Es una señal de activo bajo que se utiliza para informar al microprocesador que ha ocurrido un error de paridad en la memoria o en el dispositivo de Entrada/Salida.

RESET DRV (reset drive) : Esta señal se usa para borrar o inicializar la lógica del sistema al arrancar o en caso de que el nivel de voltaje de suministro rebase el rango de operación para mantener el funcionamiento del sistema. Esta señal se sincroniza al descender la señal OSC.

DRQ1 a DRQ3 (DMA requests) : Estas líneas de entrada son peticiones para sincronizar el canal a usar como un dispositivo periférico para tener el servicio de DMA. Una línea DRQ debe mantenerse a un nivel alto hasta que la línea correspondiente DACK cambie a nivel bajo. Tomar en cuenta que DRQ0 no se encuentra en la configuración del bus, se usa para refrescar la memoria dinámica del sistema.



DACK0 a DACK3 (DMA acknowledge signals): Estas líneas de activo bajo se usan para el reconocimiento de petición de DMA y para refrescar la memoria dinámica (DACK0).

T/C (terminal count) : Esta línea proporciona una señal en forma de pulso cuando se logra la petición de DMA.

3.2.1.

Diseño del puerto paralelo de E/S.

La IBM PC, para poder procesar datos, necesita una manera de comunicarse con el mundo exterior. El hardware de E/S y su correspondiente software proveen la interfaz para la transferencia de datos entre la computadora y un dispositivo periférico.

En este caso se utiliza una forma de transferencia de datos conocida como E/S mapeada en memoria, con la cual podemos conectar los puertos al microprocesador de tal forma que puedan ser accedidos como si fueran direcciones de memoria. Comúnmente, los puertos se conectan y son accedidos por comandos especiales del microprocesador.

La figura 3.4 muestra un diagrama a bloques de una operación básica de E/S. La lógica de decodificación genera el pulso de selección de circuito CS (Chip Select) de la dirección del bus y también genera las señales IOR y IOW. Durante el pulso de selección del puerto de entrada, se almacenan los datos provenientes del procesador en un circuito de amarre (latch). Un puerto de entrada incluye circuitos de acceso al bus triestados para aislar los datos de entrada con el bus, excepto durante el ciclo de lectura del bus. Un puerto de salida usa las memorias temporales para almacenar los datos hasta que los dispositivos de salida relativamente lentos puedan leer los datos. En la figura 3.4 los DIP switches y el display de leds sirven como dispositivos de entrada y salida respectivamente.

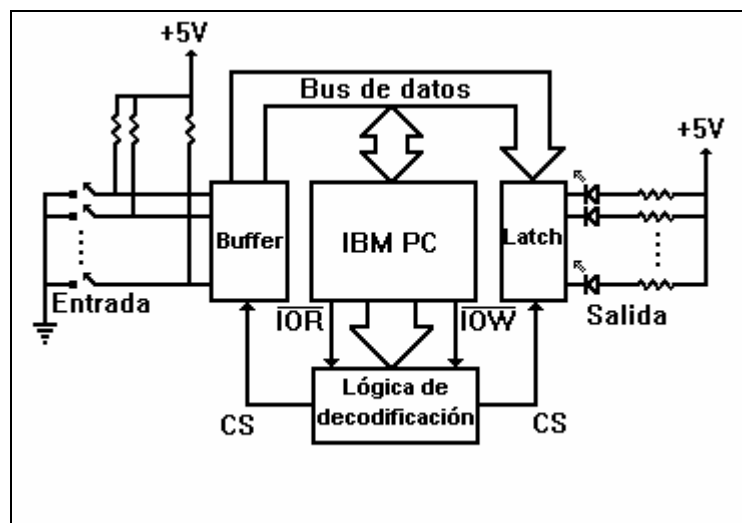


Figura 3.4. Operación básica de E/S.

3.2.2. Mapeado de E/S en la IBM PC.

Para poder generar señales de selección de circuito (CS) correctas, se deben conocer las direcciones del puerto de E/S y su mapa de asignación. El diseño de la PC proporciona 10 bits para direccionar el puerto, estos son los bits A0 a A9, para dar un total de 1024 posibles direcciones de puerto. El mapa de direcciones del puerto de E/S se divide en dos partes, las primeras 512 direcciones de 0000H a 01FFH (en notación hexadecimal) se asignan a la tarjeta madre. El espacio de direcciones de 0200H a 03FFH, las 512 direcciones restantes, son para direccionar tarjetas en los 5 slots disponibles. La tabla No. 1 muestra las direcciones de estos 512 puertos y su asignación en la IBM PC. Los puertos designados para la tarjeta son de la dirección 300H a la 31FH, que son solo 32 direcciones del puerto.

RANGO	USO	RANGO	USO
000-00F	DMA chip 8237A-5	2F8-2FF	Com. asíncronas(2)
020-021	Interrupciones 8259A	300-31F	Tarjeta de Adquisición
040-043	Reloj 8253-5	320-32F	Disco duro
060-063	PPI 8255A-5	378-37F	Impresora
080-083	Registros de página DMA	380-38C	Comunicaciones SDLC
0Ax	Registro de mask NMI	380-389	Com. Bin. Síncronas(2)
0Cx	Reservado	3A0-3A9	Com. Bin. Síncronas(1)
0Ex	Reservado	3B0-3BF	Display IBM o impresora
100-1FF	No disponible	3C0-3CF	Reservado
200-20F	Control de juego	3D0-3DF	Color / Gráficos
210-217	Unidad de expansión	3E0-3E7	Reservado
220-24F	Reservado	3F0-3F7	Disco flexible
278-27F	Reservado	3F8-3FF	Com. Asíncronas(1)
2F0-2F7	Reservado		

Tabla 1. Asignación de direcciones de puerto en una IBM PC.

3.2.3. Sincronización

En el diseño de cualquier interfaz al sistema de la PC, se debe poner especial atención en la estabilización de la compatibilidad de tiempos entre nuestro diseño y el bus de la PC. La figura 3.5 muestra la información

de tiempos que se realizan en un ciclo de bus, en los casos de lectura y escritura.

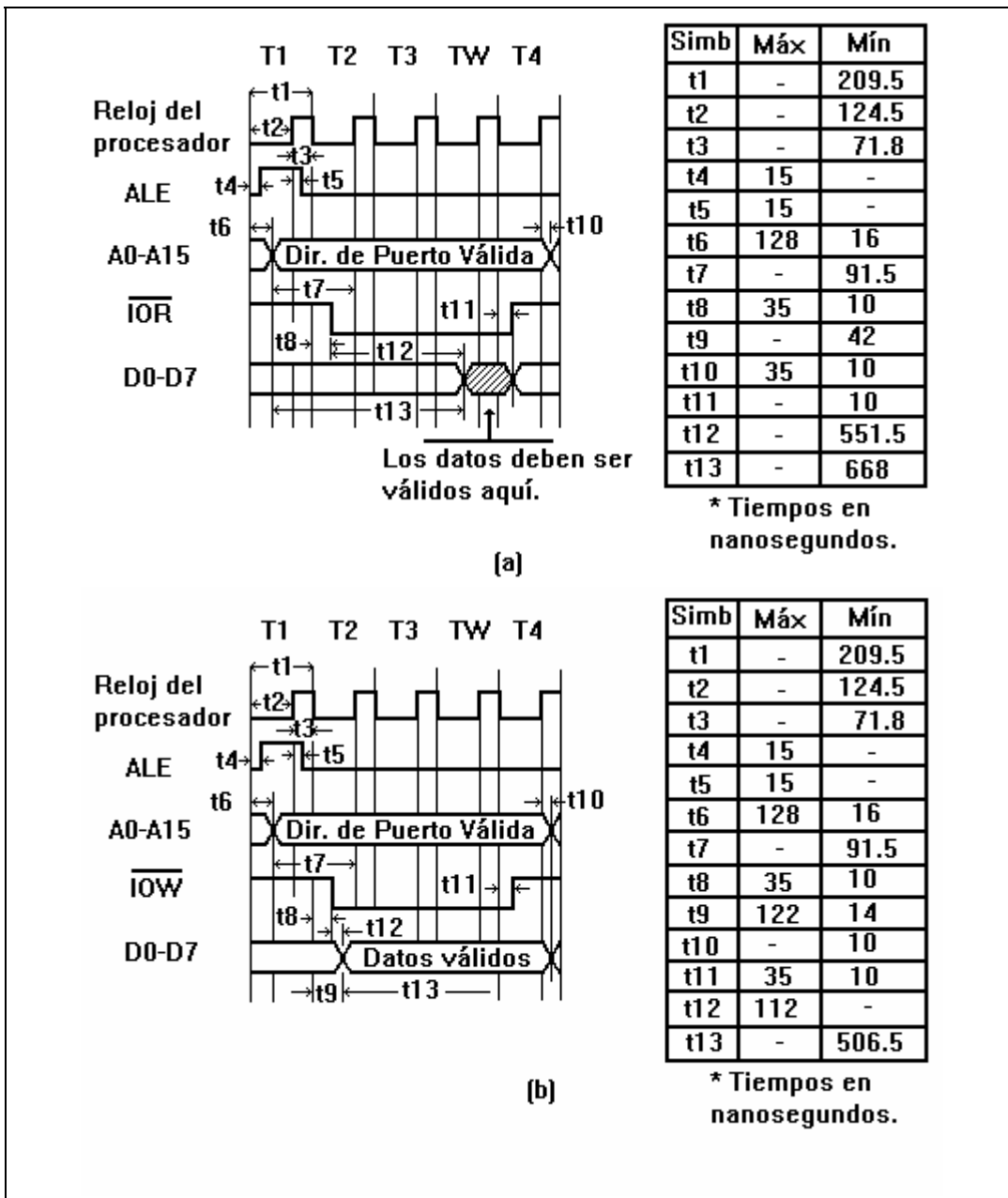


Figura 3.5. Información de tiempos en un ciclo de bus de lectura y escritura.

Estos ciclos de bus normalmente son 4 ciclos de reloj T-estado de longitud, el diseño de la PC proporciona automáticamente un periodo de reloj extra (TW) T-wait. Por lo que en la PC, todos los ciclos de bus de E/S son mínimo 5 ciclos de reloj o aproximadamente 1.05µs de longitud. Los ciclos de bus se pueden manipular controlando la señal (IO CH RDY) en el bus

del sistema. Notar que las patillas A16 a A19 de las direcciones del bus en la PC no se activan durante los ciclos de bus de E/S.

El ciclo de lectura de bus se inicia cada vez que el procesador ejecuta una instrucción de entrada. Durante T1, la señal ALE se activa, lo cual indica que los bits A0 a A15 del bus de direcciones contiene una dirección de puerto válida. Durante T2, la señal IOR del bus de control se activa, lo que indica que el puerto de entrada direccionado deberá responder, poniendo su contenido en el bus de datos. Al iniciar T4, el procesador toma los datos del bus de datos y desactiva la señal IOR.

Un ciclo de escritura se inicia cada vez que el procesador ejecuta una instrucción de salida. La señal ALE se activa, indicando que el bus de direcciones contiene una dirección de puerto válida durante T1. Durante T2 la señal IOW del bus de control se activa, de esta forma se lee el contenido del bus de datos en T3. Al inicio de T4, la señal IOW se desactiva y el procesador remueve los datos del bus.

3.2.4.

Acceso al bus y decodificación.

Existen muchas formas de acceder al bus del sistema de la PC, la figura 3.6 muestra el diseño que provee la interfaz al bus de la IBM PC con los demás circuitos utilizados en la tarjeta de adquisición. El circuito de acceso al bus, también conocido como "buffering" proporciona las direcciones BD0 a BD7 de acceso al bus de datos; las direcciones BA0 a BA9, el acceso al bus de direcciones y algunas otras señales de control, como BIOR, BLOW, BCLOCK. La primera letra B indica que es una señal de salida del circuito de acceso al bus de la PC. El circuito de decodificación genera 8 señales de selección para 8 direcciones de la tarjeta, E0 a E7 (0300H a 0307H).

Se utilizan 2 circuitos de acceso al bus tri-estado (74LS244) para acceder a los bits A0 a A9 del bus de direcciones y las señales de control necesarias en la tarjeta de adquisición. Para poder direccionar las 8 direcciones de puerto, utilizamos un circuito comparador (74LS688) para comparar el conjunto de valores 03XXH con los bits de direcciones (A5 a A9) en el bus y activar una señal de E/S (IOA) de selección de la tarjeta. Esta señal habilita el circuito bus transceiver (74LS245) y también activa la decodificación de los bits de direcciones A0 a A4. La decodificación se realiza con el circuito decodificador de 3 a 8 (74LS138). La señal BIOR controla la dirección de los datos en el bus transceiver. De igual forma que la señal de activación (CS), la señal ALE se usa para desactivar al

comparador durante una operación del ciclo de bus en la que el puerto de E/S no pueda ser accesado.

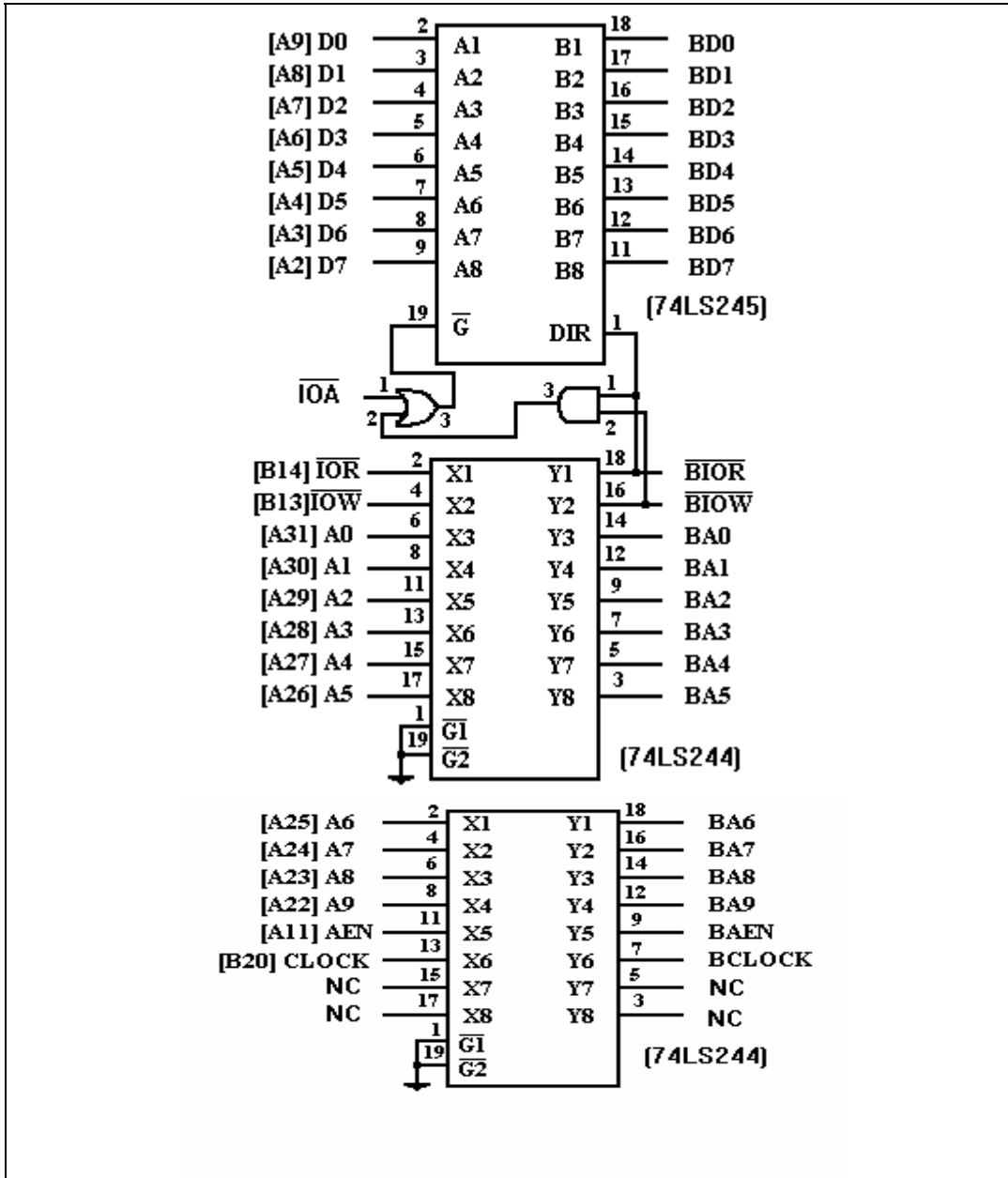


Figura 3.6.(b) Circuito de acceso al bus de la IBM PC.

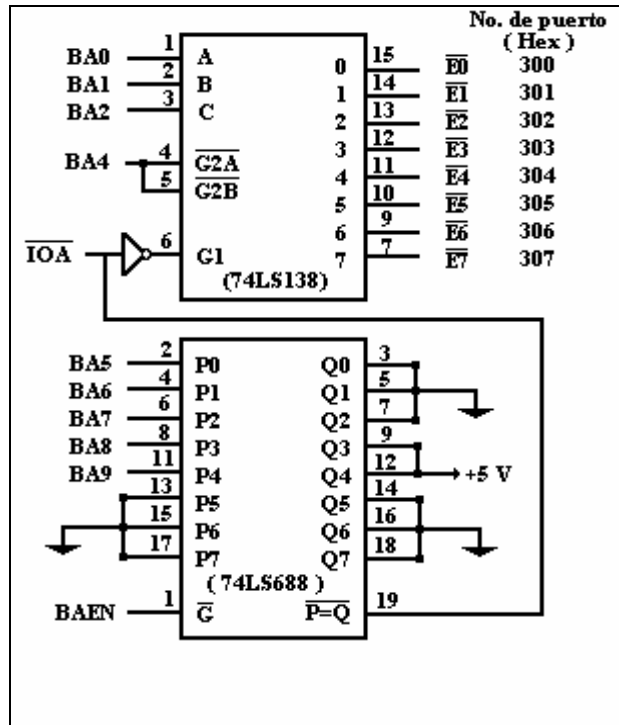


Figura 3.7. Circuito de decodificación de la tarjeta de adquisición de datos.

3.2.5. Reloj / Contador programable 8253.

Una de las características fundamentales de la mayoría de los sistemas de adquisición de datos es la forma en la que se toman o muestrean los datos, esto ocurre en intervalos de tiempo específicos, con los cuales se puede controlar el tiempo de adquisición y las muestras por segundo necesarios para que la señal pueda ser tomada con la mejor precisión y características adecuadas (capítulo 2); teniendo al final una señal a la que se puede realizar el procesamiento requerido.

Comúnmente, esta señal de reloj o de tiempo de muestreo se toma directamente del reloj interno de la IBM PC (CLOCK). También, en la mayoría de estos casos la frecuencia del reloj no es la deseada para manipular los circuitos de la tarjeta de adquisición, por lo que es necesario adaptar esta señal de reloj a la frecuencia y las características que se requieren. El sistema Analizador Dinámico de Señales, cuenta con una tarjeta de adquisición de datos, la cual controla un Convertidor Analógico a Digital (ADC0809), y no excluye este problema. Es por esto

que se utiliza un reloj programable (8253) para poder adaptar la señal de reloj a las necesidades del ADC0809.

La operación de la mayoría de los relojes es relativamente simple, como se muestra en la figura 3.8. Existen variaciones en la presentación de este esquema, pero el proceso esencial es el de tener un contador a la salida de un oscilador. Comúnmente, el oscilador opera a frecuencias altas, cuyo rango se encuentra entre los Mhz de frecuencia. El contador simplemente almacena el número de oscilaciones; usualmente actúa como un divisor de frecuencias programable, del cual se puede elegir alguna frecuencia y almacenarla en un registro para poder mantener el contador a esta frecuencia elegida. Por ejemplo, un oscilador de 1 Mhz se puede usar a frecuencias menores de 1 Mhz, con un factor de 10, 100 o 1000. La salida del divisor se envía al contador, el cual proporciona, dependiendo de su diseño, incrementos o decrementos. La mayoría de estos contadores son de 16 bits.

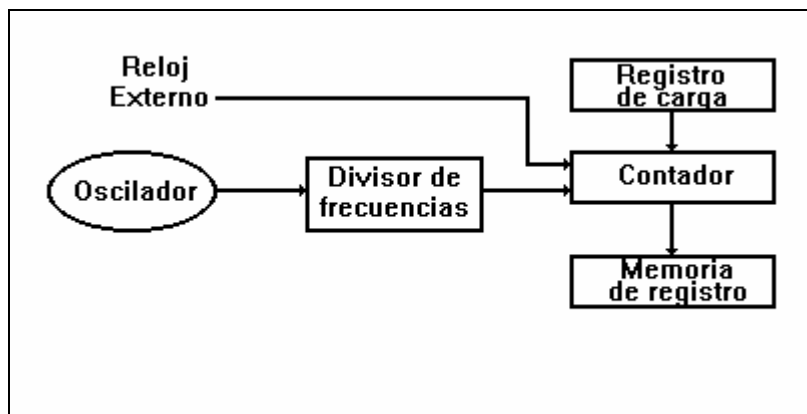


Figura 3.8. Esquema general de la operación de un reloj.

Por ejemplo, supongamos que necesitamos un reloj que tenga un disparo cada 100 microsegundos, teniendo un reloj de 10 KHz y que usamos un contador descendente. Asumiendo que se cuenta con un oscilador con una frecuencia de 1 Mhz, se necesita almacenar en el registro de carga el valor inicial de 100. El proceso de conteo se inicia al pasar el valor del registro de carga al contador. Después de 100 microsegundos de conteo, el contador leerá el valor de 0 y dejará de contar, con lo que se obtiene la señal de reloj a la frecuencia deseada, que fue la de 100 microsegundos.

El reloj puede ser manipulado por hardware o software en diferentes modos, generalmente el reloj genera una señal mientras no encuentre un 0 o algún otro valor predeterminado. En algunos sistemas, esto también es posible al generar un pulso cada vez que el reloj cambia. En otro caso, el usuario puede leer el reloj por medio de software y efectuar alguna

operación mientras el reloj no encuentre un 0. Obviamente, al manipularlo por hardware, el reloj es más exacto, ya que el retardo de lectura por medio de software puede ser considerable, este proceso básico es simple, pero puede presentar complicaciones. Por ejemplo, cuando se lee el estado del reloj, éste puede cambiar su contenido y con ello se pierde exactitud. También cuando existen cambios en los registros continuamente, por lo que es necesario utilizar una memoria del registro, para que podamos leer desde ésta, mientras el contador continúa con su trabajo. Otra desventaja que ofrecen los contadores de 16 bits, es que presentan imprecisiones cuando se trata de contar periodos relativamente largos. La solución a este problema, es la de conectar dos o más contadores en cascada. En este caso, el primer contador cuenta regresivamente hasta 0, el segundo se decrementa en uno. Este último es el que se lee por la computadora u otro dispositivo, cada contador puede tener sus propios registros de carga y de memoria.

Una tercera complicación es la necesidad de realizar el proceso de conteo repetidas veces; resulta complicado e inexacto reinicializar el proceso de conteo después de que éste termina. La forma ideal sería, tener un reloj en conteo y que indique que el proceso ha terminado, esto sería posible si los registros de carga automáticamente reinicializarán el proceso, sin embargo, cuando el contador lee un 0 (u otro número predefinido), se envía una señal eléctrica para que el usuario pueda leerla y el contador sea recargado para empezar de nuevo el conteo. Este es un modo continuo de operación, y el proceso continua hasta que se le envía una señal que le indique que pare de contar.

Otra opción, es la de contar con un generador de frecuencias en el oscilador, permitiendo con esto sincronizarlo con algún evento externo, además de que el reloj se pueda usar como un contador. En este modo, se simplifica el hecho de estar monitoreando el valor del contador para contar el número de eventos externos que han ocurrido, donde cada evento externo se mide como un pulso con las características apropiadas en la entrada del contador.

Como se puede ver en la figura 3.6, este sistema de conexión o de acceso al bus de la IMB PC cuenta con la señal de reloj de la computadora (CLOCK), la cual es de 16 MHz, para poder activar el convertidor analógico a digital, se necesita una señal de reloj, pero de frecuencia compatible al ADC0809, que es de 500 KHz. En el diseño de la tarjeta se contempla el uso de un reloj programable, que es el Intel 8253, éste cuenta con tres contadores de 16 bits independientes. Cada contador tiene su propio reloj, salida (borrow) y línea de compuerta (enable). Los contadores pueden contar en binario o en BCD, y existen 6 modos diferentes de operación:

Modo 0. Después de terminar el conteo, la salida toma un nivel alto. La señal para controlar la compuerta (GATE) debe ser de nivel bajo para el conteo y un nivel alto lo habilita.

Modo 1. Este modo es para programar un solo disparo. La salida toma un nivel bajo mientras se realiza el conteo. Cuando ocurre un disparo, en el activo alto de la señal de control de la compuerta, se habilita el conteo y borra la cuenta después de la siguiente señal reloj.

Modo 2. En este modo se produce una serie de pulsos. La salida toma un nivel alto mientras el reloj esta contando y se toma un nivel bajo para realizar un ciclo de reloj mientras termina el conteo. Este proceso se repite mientras la señal en la compuerta esté a nivel alto. Cuando se presenta un nivel bajo en la compuerta, se deshabilita el conteo y se pone la salida a nivel alto. Un pulso activo alto reinicia al contador.

Modo 3. Este modo es similar al modo 2, pero produce una forma de onda cuadrada como salida. Si el contador n es par, la salida es nivel alto para $n/2$ ciclos de reloj y es nivel bajo para $n/2$ ciclos de reloj. Si el contador es impar la salida toma un nivel alto para $(n+1)/2$ conteos y un nivel bajo para $(n-1)/2$ conteos. La compuerta se comporta de igual forma que para el modo 2.

Modo 4. La salida normalmente es nivel alto en este modo y toma un nivel bajo al realizarse un ciclo de reloj. Cuando la compuerta recibe un nivel bajo, se deshabilita el contador y un nivel alto la habilita.

Modo 5. La salida es la misma que en el modo 5, excepto que el contador se habilita cuando la compuerta recibe un activo alto.

Los relojes pueden tener cualquier frecuencia mayor a 2 MHz. Sus salidas son de gran flexibilidad y se pueden utilizar como interrupciones o relojes para otros dispositivos, en este caso funciona como un reloj programable para el convertidor ADC0809.

El 8253 puede ser activado por hardware o por software, en este caso es por software, con lo que se puede elegir el modo y el estado inicial del contador.

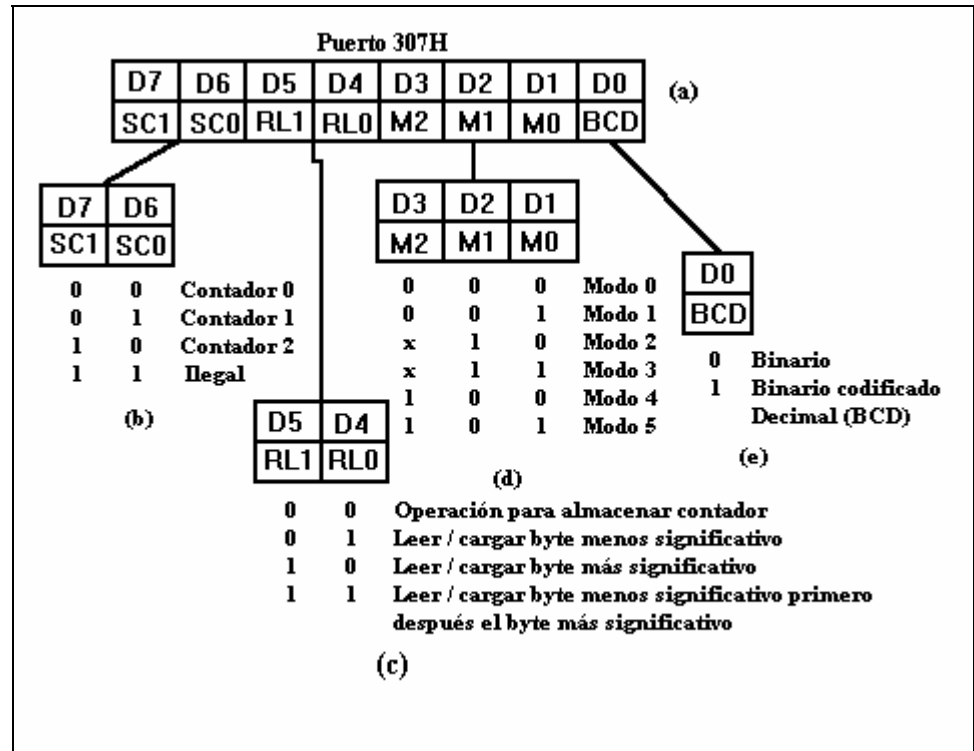


Figura 3.9. Asignación de bits de la palabra de control del 8253.

La figura 3.9(a), muestra las asignaciones de bit de la palabra de control que se debe enviar al 8253 para inicializar el contador deseado. Con los dos bits más significativos (SC1 y SC0) se elige el contador a ser utilizado. La figura 3.9(b), muestra las combinaciones para esta asignación.

Los siguientes dos bits (RL1 y RL0) llaman al 8253 cuando se requiera leer su información, cargar o contar en él. Debido a que los contadores son de 16 bits, existen tres opciones para cargar al contador. El usuario puede cargarlo (1) solo con el byte menos significativo, (2) con el byte más significativo o (3) primero el byte menos significativo y después el más significativo. Si solo se carga un byte, el otro byte se asume como cero. La figura 3.9(c) muestra exactamente estas asignaciones.

Los siguientes tres bits (M2, M1 y M0) seleccionan uno de los 6 modos de operación. La figura 3.9(d) muestra las asignaciones posibles y sus significados. Los bits menos significativos direccionan al reloj para contar a 16 bits en binario o 4 décadas en BCD, la figura 3.9(e) muestra su asignación.

Para programar el reloj, primero se envía la palabra de control al puerto designado, después se envía la cuenta inicial al puerto del reloj

apropiado. Las compuertas (GATE0, GATE1 y GATE2) se encargan del control posterior.

A veces, el usuario puede leer el estado del contador. Por ejemplo, para ver cuantos pulsos han ocurrido desde la última lectura al contador. Este método permite usar al 8253 como un contador de pulsos. Existen dos formas de leer la cuenta que lleva un contador. La primera es, deshabilitar el contador específico usando la compuerta y leer del puerto del contador. La lectura sigue el mismo procedimiento que al inicializar el contador; por ejemplo, si solo se cargó el byte bajo, solo éste puede ser leído. Si ambos bytes se cargaron, pueden ser leídos ambos, primero el byte bajo y después el alto.

Si el usuario no desea deshabilitar el contador, también puede leerlo; este método se realiza enviando una palabra de estado al contador apropiado [figura 3.9(a) y (b)] y seleccionando los bits de lectura o de carga para poner al contador en el modo de almacenamiento, como muestra la figura 3.9(c), el próximo conteo válido se carga en un registro o buffer, el cual puede leer el usuario de la forma descrita en el método anterior.

3.2.5.1. Conexión del 8253.

El sistema de la IBM PC cuenta con un 8253, pero también utiliza otros tres relojes. Uno para refrescar la memoria, otro para el generador de tonos y el tercero para la interrupción de la hora y fecha del día. Esta última interrupción provee al usuario con 18.2 interrupciones/seg. que son utilizadas la mayoría de veces por el software en la localidad de memoria 001CH. Sin embargo, para otras aplicaciones este es un rango muy lento.

La figura 3.10 muestra el esquema del circuito 8253 que fue implementado en el diseño de esta tarjeta de adquisición. La fuente de alimentación de 5 V alimenta al circuito entre Vcc y GND. Las líneas D0 a D7 conectan al 8253 con el bus de datos de la IBM PC. Las líneas RD y WR se conectan a las líneas del bus IOR y IOW. Las líneas A0 y A1 se conectan al bus de direcciones de la IBM PC. La tabla 2 muestra cómo se permite el direccionamiento de la palabra de estado y de los tres relojes. La selección de la señal CS se debe activar por cuatro puertos separados desde que la palabra de control y los contadores son direccionados por separado. Las líneas de decodificación E0 (puerto 300H) a E3 (puerto 303H) accesan dichos puertos, mientras que la línea E4 (puerto 304H) controla las compuertas del 8253 con ayuda del circuito de amarre 74LS373 para permitir que la señal de salida sea la patilla OUT2. Ponemos

el contador 0 a 1.1925 Mhz, que es una cuarta parte de la frecuencia del reloj del sistema de la IBM PC, usando el circuito de la figura 3.11. Los tres contadores están conectados en cascada para proporcionar un conteo de 48 bits, con esto, dividimos la frecuencia del reloj de entrada a $2^{48} - 1$. También en esta figura se muestra la configuración del dipswitch para poder utilizar la tarjeta en computadoras más rápidas.

La tabla 3 muestra el puerto y la asignación de los bits para el 8253. En este caso, el 8253 se configura para funcionar como un contador, conectando los tres contadores en cascada, se pueden habilitar al mismo tiempo y mantienen en su registro la cuenta inicial.

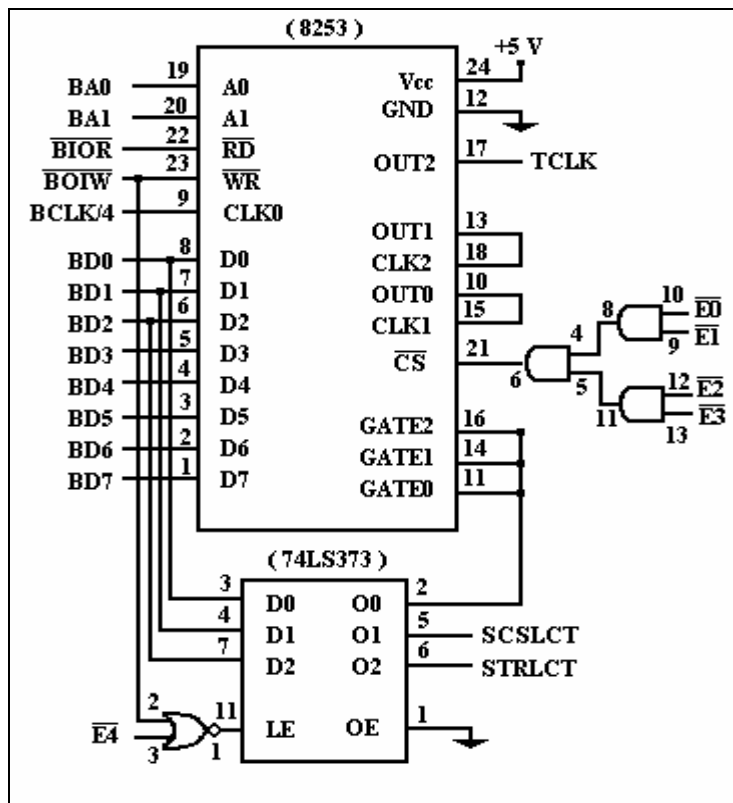


Figura 3.10. Diagrama de conexión del 8253.

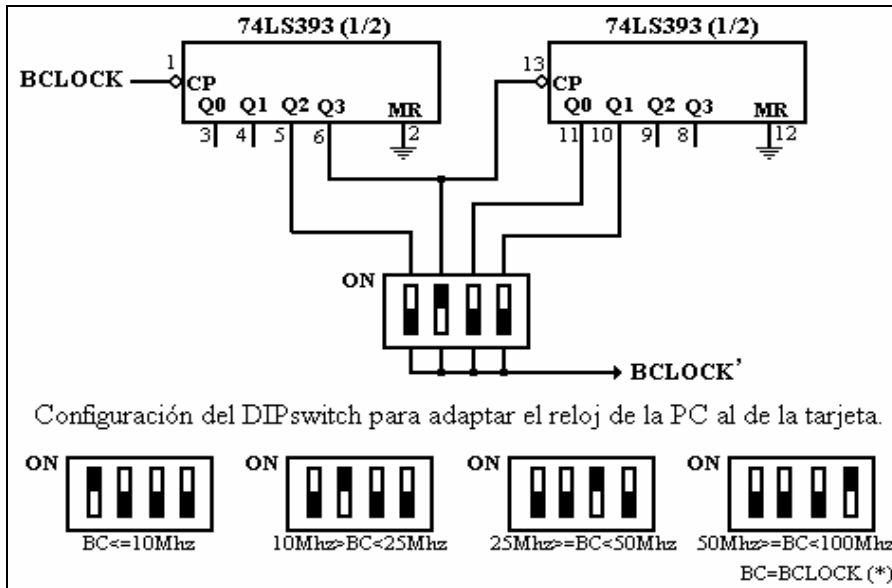


Figura 3.11. Diagrama del circuito usado para dividir la frecuencia de la señal de reloj de la computadora.

RD	WR	A1	A0	
1	0	0	0	Cargar contador 0
1	0	0	1	Cargar contador 1
1	0	1	0	Cargar contador 2
1	0	1	1	Escribir palabra de control
0	1	0	0	Leer contador 0
0	1	0	1	Leer contador 1
0	1	1	0	Leer contador 2

Tabla 2. Direccionamiento de los contadores y del estado de la palabra de control.

Dirección de puerto (Hex)	
304	Contador 0
305	Contador 1
306	Contador 2
307	Registro de control
30C	Puerto de compuerta (solo bit 0)

Tabla 3. Asignación de puertos para el 8253.

La figura 3.12 muestra el código en lenguaje C++ de la función que inicializa al 8253 para producir una secuencia de pulsos a una frecuencia dada por los parámetros de entrada especificados por el usuario. La función Reloj(periodo, unidad) cuenta con los parámetros (periodo) que es el periodo de tiempo entre los pulsos y (unidad) que es la unidad de tiempo. Los posibles valores para (unidad) son 's', 'm' o 'u' los cuales seleccionan el periodo en segundos, milisegundos o microsegundos respectivamente. Los microsegundos se redondean al múltiplo de 3.3545 μ s más cercano. Por ejemplo, el comando: Reloj(5 , 'm'); indica que deseamos producir una cadena de pulsos con un periodo de 5 ms entre cada pulso. Con esto podemos programar al 8253 para producir la señal necesaria para realizar la conversión analógica a digital.

```

#define CNT0 0x300
#define CNT1 0x301
#define CNT2 0x302
#define TCNTRL 0x303
Reloj(periodo,unidad)
int periodo;
char unidad;
{
    char tic[6];
    unsigned int tottics;
    if(unidad=='u')
    {
        tottics=(float) periodo/3.3543;
        tic[5]=tottics/256;
        tic[4]=tottics%256;
        tic[3]=0;
        tic[2]=2;
        tic[1]=0;
        tic[0]=2;
    }
    if(unidad=='m')
    {
        tic[5]=(periodo*2)/256;
        tic[4]=(periodo*2)%256;
        tic[3]=1;
        tic[2]=42;
        tic[1]=0;
        tic[0]=2;
    }
    if(unidad=='s')
    {
        tic[5]=(periodo*2)/256;
        tic[4]=(periodo*2)%256;
        tic[3]=0x17;
        tic[2]=0x4B;
        tic[1]=0;
        tic[0]=100;
    }
    outportb(TCNTRL,0x34);/*contador 0, carga LSB y MSB, modo 2*/
    outportb(TCNTRL,0x74);/*contador 1, carga LSB y MSB, modo 2*/
    outportb(TCNTRL,0xB4);/*contador 2, carga LSB y MSB, modo 2*/
    outportb(CNT0,tic[0]);
    outportb(CNT0,tic[1]);
    outportb(CNT1,tic[2]);
    outportb(CNT1,tic[3]);
    outportb(CNT2,tic[4]);
    outportb(CNT2,tic[5]);
}

```

Figura 3.12. Código en lenguaje C++ de la función Reloj(periodo,unidad).

La función Reloj(periodo , unidad) pone los tres contadores a operar en el modo 2, produciendo una cadena continua de pulsos repetitivos, es importante que los registros de los contadores se carguen con una cuenta inicial menor a 2 para poder producir la salida. El efecto a esto es dividir la cuenta inicial en 2. En general, la función Reloj(periodo , unidad) se diseñó para cargar los primeros dos registros del 8253 para producir una entrada de reloj al último contador que podrá ser la mitad de la unidad de tiempo especificada. La mitad de la unidad especificada se hace al cargar un valor de 2 en el segundo registro del contador. Esto produce pulsos de duración igual a la mitad de la unidad de tiempo para poner la salida CLK2 a la frecuencia especificada. Debido al uso de la tableta de pruebas (protoboard), el periodo mínimo que podemos producir entre los pulsos TCLK es de $3.354 \mu\text{s}$. Esto se almacena al cargar cada uno de los tres registros de los contadores con el número 2. Por lo que cuando se usa la unidad de tiempo de microsegundos, la salida se ajustará al múltiplo más cercano de $3.354 \mu\text{s}$.

3.2.6.

Conexión del ADC0809.

En esta sección, se analiza el diseño del sistema de adquisición de datos para la IBM PC. El objetivo fue obtener un sistema de adquisición de datos a un bajo costo, teniendo en cuenta que pueda ser construido por cualquier usuario o estudiante de electrónica. El diseño se basó en el convertidor analógico a digital ADC0809 (National Semiconductor), el cual es un convertidor monolítico de 8 bits. Este circuito fue seleccionado ya que, se puede adquirir a un bajo costo, es de tipo multicanal y cuenta con un circuito multiplexor analógico para poder seleccionar el canal con el que deseamos realizar la conversión.

Una de las aplicaciones más usadas del ADC0809 es muestrear una señal a una frecuencia constante (por ejemplo 1000 muestras / seg.). Lo cual requiere una señal de reloj para manejar la línea de inicio de conversión del ADC.

La figura 3.13 muestra el diagrama del circuito para el ADC. En él ponemos sus conexiones con el sistema de bus y el circuito de decodificación, estos últimos analizados anteriormente. El ADC cuenta con 8 líneas de entrada con un rango de voltaje de 0 a 5V. para operar este convertidor, conectamos las líneas para selección del canal en los 4 bit menos significativos de un byte de datos y con un comando de software se envía el byte al puerto 305H, el cual selecciona las líneas E5 y BLOW, seleccionando así el canal, ya que los 4 bits menos significativos se almacenan en registro de dirección de canal del ADC. La línea SCSLCT (

Start Conversión SeLeCT) se almacena por medio del puerto 304H (ver figura 3.7) y controlando la conversión al inicializar con la línea E4, la cual activa al reloj y habilita al circuito 74LS126 para enviar la señal de reloj CLOCK. La señal de ALE al ADC0809; ésta misma señal activa en su activo descendente a la línea START del ADC0809, dando como resultado el inicio de la conversión. Si la señal SCSLCT está en un nivel alto, cualquier comando de entrada o de salida con cualquier argumento de datos activa la selección del puerto 305H (correspondiente a la línea E5) dando inicio a la conversión; si la señal SCSLCT es bajo, se desactiva el paso de las señales, de reloj, la de ALE y START hacia el ADC0809, con lo cual se da fin a la conversión; teniendo ya en la memoria de la PC las muestras tomadas por el ADC0809. Durante este lapso, es necesario retroalimentar la señal de START para volver a reiniciar una conversión, el ADC0809 cuenta con una línea de salida EOC (End Of Conversion), la cual indica que ha finalizado la conversión. La figura 3.13 muestra el circuito que realiza la labor de retroalimentar la señal de START, ya sea por medio de la línea E5, correspondiente al puerto 305H o por medio de la señal de EOC, esto solo si la señal SCSLCT se encuentra a un nivel alto.

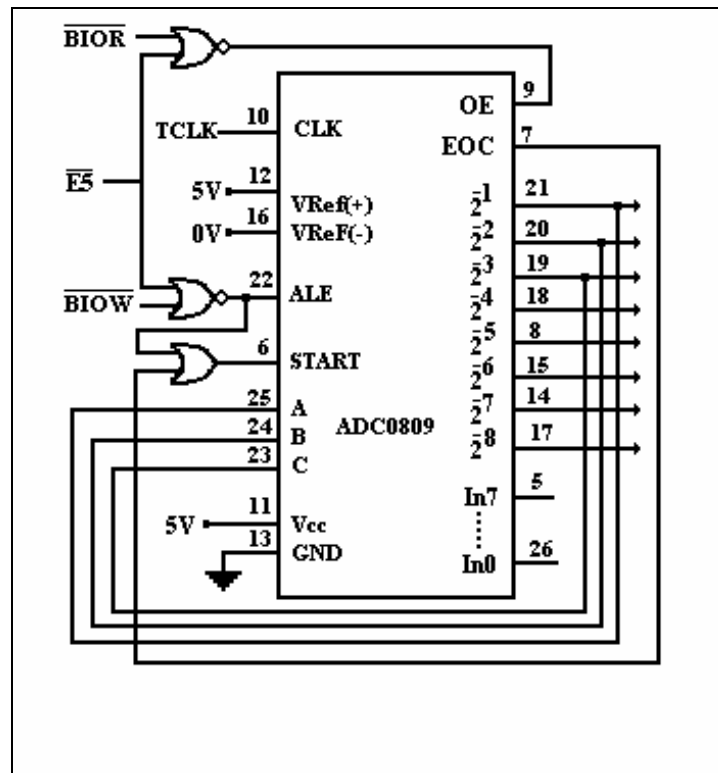


Figura 3.13. Diagrama de conexión del ADC0809.

La señal de EOC (End Of Conversion) toma un nivel bajo cada que el convertidor termina una conversión, con lo que la señal de EOC se puede

utilizar como reloj para realizar lo que se conoce como "polling", es decir, para realizar una conversión, la señal de inicio (START) inicia la conversión y el circuito de la figura 3.13 se encarga de monitorear la señal de EOC para utilizarla como un reloj retroalimentador y repetir el inicio de conversión secuencialmente, este mismo circuito cuenta con la señal de parada (Stop) la cual se deshabilita cuando la computadora envía una señal de que ya realizó las lecturas necesarias. La computadora, por medio de la orden Inportb (dirección) selecciona al puerto 305H el cual corresponde a la línea E5 y también selecciona la señal BIOR para realizar la lectura al convertidor por medio del bus de la PC.

A continuación se muestra el diagrama completo de la tarjeta de adquisición de datos, que incluye las figuras 3.2, 3.3, 3.6, 3.7, 3.10, 3.11 y 3.13 analizadas anteriormente.

3.3.

Diseño del software.

En esta sección, se mencionan los pseudocódigos que se utilizan en el programa del Analizador Dinámico de Señales, así como también se presentan los diagramas de flujo y una explicación de cada uno de los procedimientos para manipular la interfaz gráfica del Analizador Dinámico de Señales.

La programación está basada en la técnica de "entrada y salida programada", en la cual el procesador es programado para la tarea de transferencia de datos. Los programas para la computadora son las instrucciones que ejecuta la máquina para llevar a cabo la tarea encomendada.

Para lograr la captura y presentación gráfica de señales eléctricas se diseñan programas en lenguaje de bajo o de alto nivel. El programa para la presentación de resultados del Analizador Dinámico de Señales esta basado en un lenguaje de alto nivel como es el lenguaje C.

El programa esta organizado en dos tipos:

- a) programas para tareas generales.
- b) programas para tareas específicas.

Los programas para tareas generales se encargan de la comunicación con el usuario, manejar la interfaz y recursos de la máquina, así como también la presentación de los resultados en forma gráfica. Estos programas estan diseñados en lenguaje C++ y comúnmente utilizan a los programas para tareas específicas.

Los programas para tareas específicas, esencialmente son subrutinas y su principal objetivo es la realización de la captura de las señales eléctricas.

Ambos tipos de programas trabajan de manera conjunta y forman la programación general para el proceso de captura de las señales eléctricas y su representación gráfica. El diagrama de flujo para la programación general se muestra en la figura 3.14.

Aunque parece simple, este diagrama de flujo presenta en cada etapa otro diagrama de flujo específico. A continuación se desarrolla cada una de las etapas del diagrama de flujo para la programación general.

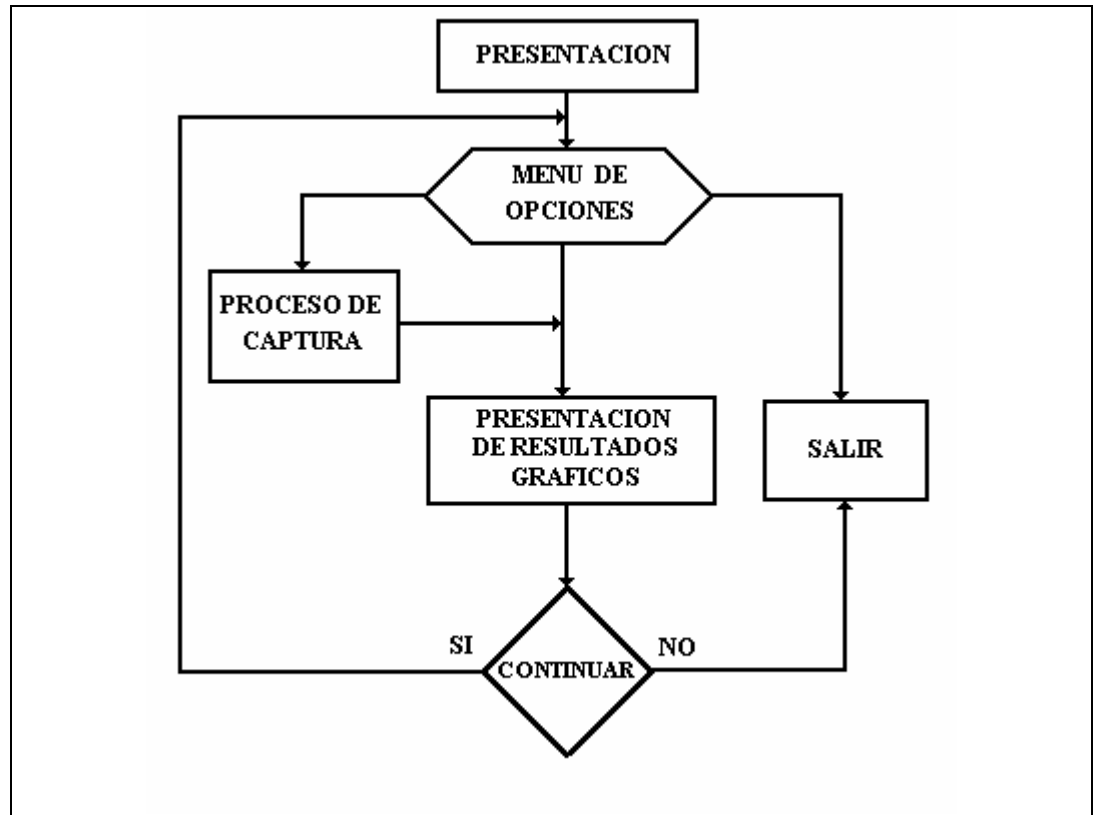


Figura 3.14. Diagrama de flujo del programa principal.

El pseudocódigo del programa principal, es el siguiente:

Programa principal

Declaración de variables globales.

inicio

```

  ch<-0
  inicia_modos_grafico( )
  si (mouse) entonces iniciar_mouse( )
  escribir texto_entrada( )
  escribir interfaz_grafica( )
  mientras (ch!=salir) hacer
    ch=-1
    si (activación_tecla) entonces
      leer ch
      llamar_a activar_subrutina( ch )
  fin_mientras
  cerrar_graficos( )

```

fin


```
función activar_subrutina( ch )
inicio
  si ch=F1 entonces llamar_a ayuda( )
  si ch=ALTG entonces llamar_a salvar( )
  si ch=ALTA entonces llamar_a cargar( )
  si ch=ALTF entonces llamar_a frecuencia( )
  si ch=ALTU entonces llamar_a muestreo( )
  si ch=ALTL entonces llamar_a valores( )
  si ch=ALTI entonces llamar_a imprime( )
  si ch=ALTS entonces llamar_a shell( )
  sino regresar al programa principal.
fin
```

3.3.1. Funciones.

Ahora se especificarán cada uno de los bloques que conforman el diagrama de flujo general, iniciando con el de menú de opciones, contamos con otros diagramas de flujo para cada una de las opciones que se presentan. En la figura 3.15 se muestra un diagrama que incluye las opciones principales en las que se divide el menú principal, a continuación se analizarán cada una de ellas.

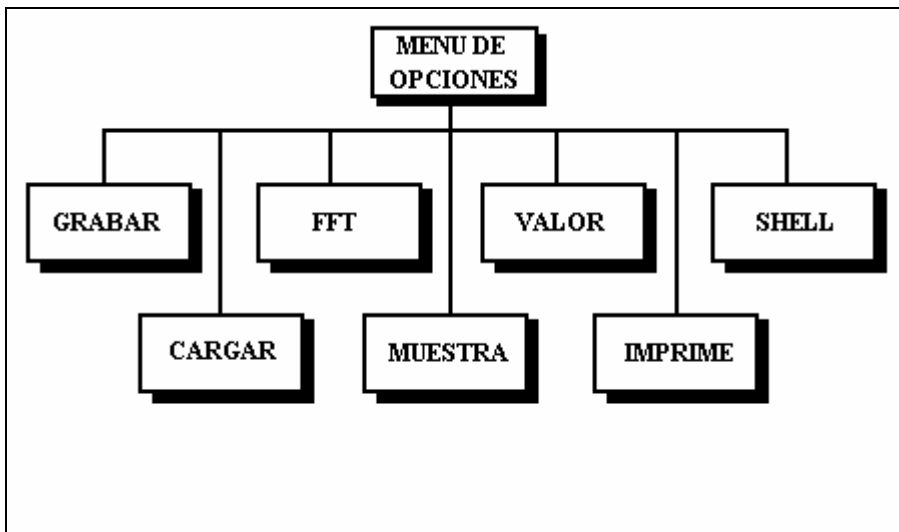


Figura 3.15. Diagrama a bloques del menú de opciones.

Grabar.

Esta subrutina se utiliza para almacenar en disco, la señal que se localiza en la memoria de la computadora. El diagrama de flujo de la figura 3.16 muestra el seguimiento de esta opción.

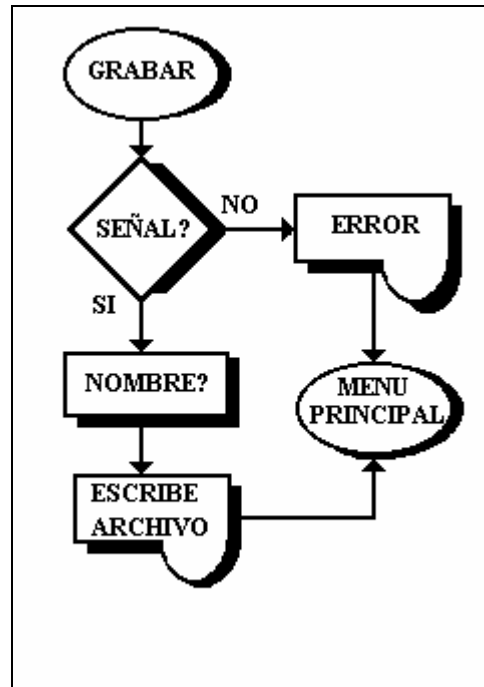


Figura 3.16. Diagrama de flujo de la función grabar.

El pseudocódigo es:

```
función grabar( )
inicio
    leer nombre_archivo
    si ( escribe nombre_archivo ) entonces
        escribe archivo
    sino escribe error
fin
```

Cargar.

Esta subrutina se usa para cargar algún archivo que contenga alguna señal muestreada por el ADS. La señal se carga en el canal seleccionado. Su diagrama de flujo es el que muestra la figura 3.17.

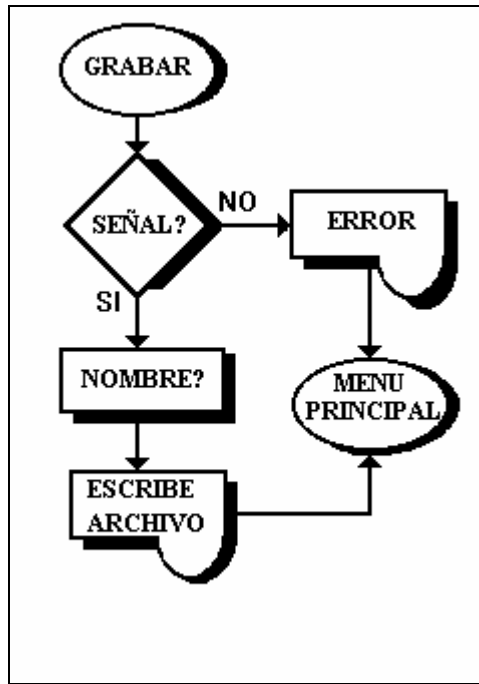


Figura 3.17. Diagrama de flujo de la función cargar.

El pseudocódigo es:

```

función cargar( )
inicio
    leer nombre_archivo
    si ( nombre_archivo ) entonces
        leer archivo
        mostrar_señal_gráfica
    sino escribe error
fin
  
```

FFT.

Esta subrutina es la que calcula la Transformada Rápida de Fourier a la señal que se encuentra en la memoria de la computadora. El resultado se almacena en la misma computadora, siendo el espectro de la señal. Estos resultados se visualizan en la ventana del dominio del tiempo.

El pseudocódigo es:

```

función frecuencia( )
var muestras, potencia, imaginarios[]
inicio
    si (señal) entonces
        imaginarios<-0
        muestras<-MAX
  
```

```

        potencia<-log(MAX)/log(2)
        desde i=0 hasta MAX hacer
            señal/=muestras
            imaginarios/=muestras
            i<-i+1
        fin_desde
        llamar_a FFT( )
        mostrar_senál_gráfica
    sino escribe error
fin

función FFT( )
var i1, i2, i3, i4, y, ciclo1, ciclo2, a1, a2, b1, b2, z1, z2, v,
index, resultado
inicio
    i3<-0
    i4<-i1
    v<-6.283185*(1/MAX)
    desde ciclo1<-0 hasta MAX hacer
        index<-i3/i1
        desde i=0 hasta potencia hacer
            n1>>=1
            si (!index<n1) entonces
                resultado+=pow(2,i)
                index-=n1
            fin_desde
            y<-resultado
            z1<-cos(v*y)
            z2<--sin(v*y)
            desde ciclo2<-i3 hasta i4 hacer
                a1<-señal[ciclo2]
                a2<-imaginarios[ciclo2]
                b1<z1*señal[ciclo2+i1]z2*imaginarios[ciclo2+i1]
                b2<-z2*señal[ciclo2+i1]+z1*imaginarios[ciclo2+i1]
                señal[ciclo2]<-a1+b1
                imaginarios[ciclo2]<-a2+b2
                señal[ciclo2+i1]=a1-b1
                imaginarios[ciclo2+i1]<-a2-b2
                ciclo2<-ciclo2+1
            fin_desde
            i3+=(i1<<1)
            i4+=(i1<<1)
            ciclo1<-ciclo1+1
        fin_desde
    desde i=0 hasta MAX hacer
        desde j=0 hasta power hacer
            n1>>=1
            si (!index<n1) entonces
                resultado+=pow(2,j)
                index-=n1
            fin_desde
        n<-resultado
        espectro[i]<sqrt(señal[i]*señal[i]+imaginarios[i]*
            imaginarios[i])
    fin_desde
fin

```

Muestra.

Esta subrutina se encarga de realizar el muestreo por medio de la tarjeta de adquisición de datos, los resultados se muestran en la ventana del dominio de tiempo. Por medio de la función Reloj(periodo , unidad), figura 3.12, se controla el reloj para inicializar el ADC0809, se controla también la entrada/salida hacia el puerto de la tarjeta.

```

función muestra( )
var canal
inicio
    leer canal
    outportb( 0x304 , 7 )
    llamar_a Reloj( 2 , "u" )
    outportb( 0x305 , canal )
    desde I<-0 hasta MAX hacer
        señal[i]=inportb( 0x305 )
        i<-i+1
    fin_desde
    outportb( 0x304 , 0 )
    mostrar_senál_gráfica
fin

```

Valores.

Esta subrutina se encarga de mostrar los valores de la señal, con la ayuda del mouse, se posiciona en la parte de la señal que se desea conocer sus valores y los muestra por medio de una ventana.

El pseudocódigo de esta función es el siguiente:

```

función valores( )
var coordenada_x, coordenada_y
inicio
    si (señal) entonces
        mientras (!(leer tecla) ) hacer
            si (!botón_izq_mouse) entonces
                ventana_tiempo
                leer coordenada_x
                leer coordenada_y
                escribir coordenada_x
                escribir coordenada_y
            sino
                ventana_frecuencia
                leer coordenada_x
                leer coordenada_y
                escribir coordenada_x
                escribir coordenada_y
            fin_si
        fin_mientras
    sino escribe error
fin_si

```

fin

Imprime.

Esta función se encarga de mandar la salida a una impresora gráfica de matriz de puntos, detecta si la impresora está lista, de otra forma envía un mensaje de error. Esta opción solo imprime las ventanas del dominio del tiempo y del dominio de la frecuencia.

```

función imprime( )
var i, j, cociente, resto, modo=0
inicio
  si (señal) entonces
    _AH<-2
    _DX<-0
    generainterruptión( 0x17 )
    si ( _AH&0x80 ) entonces
      escribe "Ms/div", "Volts/div", "T.m.", "Canal"
      escribe ( stdprn , "\x1B\A7 )
      resto<-453&0x00FF
      cociente<-453>>8
      desde j<-5 hasta 548 hacer
        escribe ( stdprn , "\x1B*%c%c%c , modo,
        resto, cociente )
        desde i<-475 hasta 22 hacer
          m<-0
          desde k<-0 hasta 8 hacer
            m<<=1
            si ( existe_pixel( j+k,i )
            entonces
              m++
            fin_si
            k<-k+1
          fin_desde
        i<-i-1
      fin_desde
      escribe (stdprn , "\x0D\x0A" )
      j<-j+8
    fin_desde
  sino escribe error
fin

```

Shell.

Esta función es la que se encarga de interactuar con el prompt del sistema operativo, cuando se activa esta función, el ADS queda en memoria y se presenta el prompt del sistema operativo, con lo cual se puede realizar cualquier tarea desde DOS, por ejemplo copiar archivos, renombrarlos, etc. Regresa al ADS cuando se tecléa EXIT y <ENTER>.

```
función shell( )  
var firts  
    cierra_modo_gráfico  
    escribe "ADS Versión 1 Type "EXIT" para  regresar a ADS")  
    putenv("HAVE_ADS=YES")  
    system("")  
    inicia_modo_gráfico  
    inicia_mouse  
    muestra_interfaz_gráfica  
    firts<-0  
    si (señal) entonces  
        muestra_señal_gráfica  
    fin_si  
fin
```

4

Conclusiones.

El sistema " Analizador Dinámico de Señales (ADS) " es el resultado de la implementación de herramientas hardware y software en un mismo sistema; apoyándose en la estructura de un instrumento virtual y de herramientas matemáticas tales como los algoritmos de la Transformada Rápida de Fourier (FFT). Mostrando que la tecnología en su conjunto está dispuesta a contribuir con herramienta propia a objetivos específicos. Los resultados son una muestra del potencial que un sistema de esta clase nos ofrece, todo lo anterior ha venido ocurriendo con la ayuda de la computadora, que como en este trabajo se explica, es una herramienta cuya flexibilidad supera cualquier obstáculo y

su área de trabajo se ha extendido a todas las ramas de la ciencia.

El resultado se puede apreciar al trabajar con el sistema ADS en aplicaciones de análisis de señales, además de que cumple con los objetivos establecidos; con la evolución de herramientas de instrumentación, tradicionales y virtuales, proporciona sofisticación y flexibilidad, El costo para adquirir algunos de estos instrumentos es alto, el Analizador Dinámico de Señales, se puede construir a un costo relativamente bajo, tomando en cuenta que sus componentes electrónicos son muy conocidos y se pueden adquirir fácilmente. Los requisitos para usarlo son; contar con una computadora con las especificaciones requeridas, ajustarlo a sus necesidades y utilizarlo como plataforma para otras aplicaciones.

Este trabajo cumple el objetivo de proporcionar el ADS a instituciones educativas que carezcan de recursos para la adquisición de herramientas de instrumentación, dando con ello, pauta para una extensa aplicación del ADS, y como base para la creación de nuevos instrumentos virtuales para resolver las necesidades que se presenten en cada aplicación.

Una de las características del ADS es dar un ejemplo de la aplicación de las herramientas matemáticas para el análisis de señales, esto es de gran importancia, ya que el ADS puede simular en un futuro a otras herramientas, como son la implementación de filtros, ejemplos de convolución, etc. La aplicación al ADS mas cercana es el desarrollo por medio de software de un generador de funciones, con lo cual el software se puede utilizar sin la tarjeta de adquisición de datos; esto es un claro ejemplo de la flexibilidad que puede proporcionar un instrumento virtual.

La computadora como un sistema, debe ser tomado en cuenta en aplicaciones no solo de ejecución de paquetes, sino como una herramienta más en la solución de problemas que se presenten. La creación de sistemas en base a una computadora empieza a ser uno de los grandes frutos de la tecnología en las áreas de investigación, automatización y control en los procesos industriales y en la educación es de fundamental importancia utilizar esta herramienta como plataforma para el cumplimiento de sus objetivos.

Bibliografía.

1. Grossman, Morris; **Test and measurement**, IEEE SPECTRUM; January 1993.
2. Byers, T. J.; **PC BASED TEST EQUIPMENT**, Electronics Now, October 1993.
3. Tompkin, Willis J., Webster, John G.; **Interfacing sensors to the IBM® PC**, Prentice Hall 1988.
4. Oliver and Cage, **Electronic measurements and Instrumentation**, Inter- university Electronics series 1971.
5. Potter W., Ronald; **SIGNAL ANALYSIS BY DIGITAL TECHNIQUES**, Hewlett Packard Company; 1987.
6. National Instruments; **DATA ACQUISITION SEMINAR**, January, 1995.
7. Oppenheim, Alan V., Shafer, Ronald W.; **Discrete-Time signal Processing**, Prentice Hall 1989.
8. Oppenheim, Alan V., Willsky Alan S.; **SEÑALES Y SISTEMAS**, Prentice Hall 1994.
9. National Instruments; **Data Acquisition Databook**, 1993.

- 10.Santori, Michael; **An instrument that isn't really**, IEEE Spectrum; August 1990.
- 11.Cerchione, Robert G.; **Data acquisition made simpler**, IEEE Spectrum; November 1992
- 12.Gates, Stephen C., Becker, Jordan; **LABORATORY AUTOMATION using the IBM® PC**, Prentice hall 1990.
- 13.Elabdalla, Abid M., Abu-El-Haija, Ahmad Y.; **Personal Computer Flexible Multichannel Interface for Data Acquisition of Low Frecuency Signal**; IEEE Transactions on Instrumentation and Meaurement Vol 37 No. 4, Dec. 1988.
- 14.Liu, C. L., Liu, Jane W. S.; **Linear System Analysis**, Mc. Graw Hill International, Student Edition; 1990
- 15.Bruce, A. Carlson; **COMMUNICATIONS SYSTEMS An Introduction to signals and noise in Electrical Communication**; Mc. Graw Hill, 1986.
- 16.Wolf, Stanley; Smith, Richard F. M. **Guía para Mediciones Electrónicas y Prácticas de Laboratorio**; Prentice Hall 1992.
- 17.Couglin, Robert F.; **Amplificadores operacionales y circuitos integrados lineales**; Prentice Hall, Cuarta edición.
- 18.Embree, Paul M. Kimple, Bruce; **C Language Algorithms for Digital Signal Processing**; Prentice Hall 1991
- 19.Zimmerman, Scott, Zimmerman Beverly B.; **La biblia del TURBO C, Fundamentos y Técnicas avanzadas de programación**; ANAYA MULTIMEDIA S.A. EDICIONES, 1990.
- 20.Ezzell, Ben; **PROGRAMACION DE Gráficos en Turbo C++**; Addison Wesley 1993.
- 21.Joyanes Aguilar, Luis; **Fundamentos de programación, algoritmos y estructuras de datos**; Mc. Graw Hill 1992
- 22.Brigham, E. O.: **The Fast Fourier Transform and Its Aplications**, Prentice Hall 1988.

Apéndice A.

Manual de usuario.

En este apéndice se mencionan las herramientas con las que cuenta el software del Analizador Dinámico de Señales, se presentan ejemplos de como se puede manipular la interfaz gráfica para obtener el análisis de señales; se presenta una explicación de cada uno de los procedimientos para interactuar con la interfaz gráfica del Analizador Dinámico de Señales. Se trata de que esta sección se utilice como manual de usuario para la utilización del software.

Instalación y configuración.

El primer paso para poder ejecutar el software del ADS es instalar los archivos en el disco duro, teniendo en cuenta los requerimientos mínimos necesarios que debe cumplir la computadora. Los requisitos son:

- Computadora IBM PC o compatible con procesador 286 o superior (coprocesador matemático recomendado).
- 4 Megabytes de RAM.
- Monitor VGA.
- Disco Duro.
- Sistema Operativo MS-DOS Versión 6.0.
- Mouse.

El ADS cuenta con un programa llamado instala.bat, el cual al ejecutarse, instala el software del ADS en la unidad que se desee, creando el subdirectorio llamado ADS por omisión, se puede elegir cualquier nombre, y copia los siguientes archivos:

- ADS.EXE [archivo ejecutable]
- MOUSE.COM [archivo para activación del mouse]
- CHAN1.ADS, CHAN2.ADS, LOG1A.ADS, LOG1B.ADS [archivos de uso temporal por el software del ADS]
- También incluye algunos archivos con extensión .SIG, los cuales son archivos de señales muestreados por el sistema ADS.

Después de la instalación, solo se necesita teclear ADS <ENTER> para ejecutar el software del sistema Analizador Dinámico de Señales.

Guía básica.

En esta parte se da una guía para el usuario, con algunos ejemplos y definiciones de las funciones con las que cuenta el software del ADS. Se da una explicación de cada una de éstas y cómo se pueden activar, ya sea por medio del mouse o desde el teclado.

Presentación.

La presentación es un texto en el cual se menciona el nombre del sistema, se requiere que se active alguna tecla para que este mensaje cambie al menú de opciones.

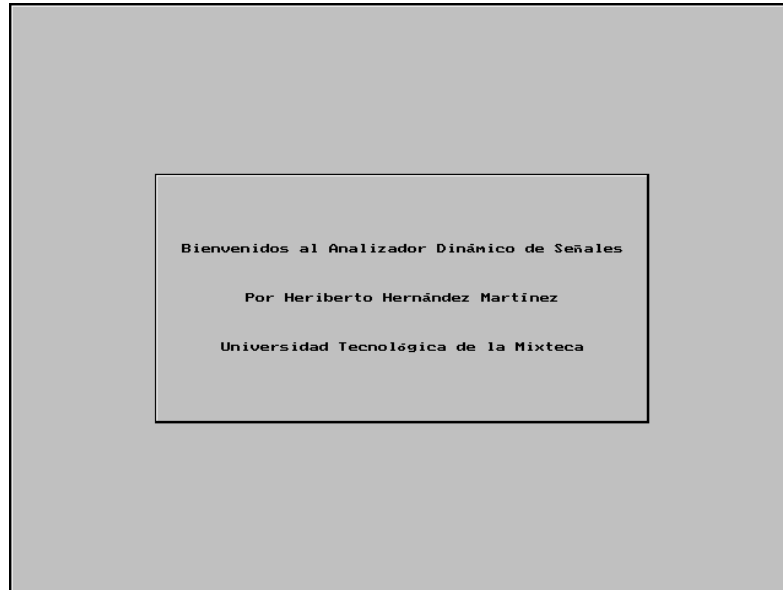


Figura A-1 Interfaz de presentación.

Menú de opciones.

En esta etapa se indican las opciones que se tienen para que el sistema inicie su trabajo, la figura A-2 es la interfaz gráfica que presenta el menú de opciones.

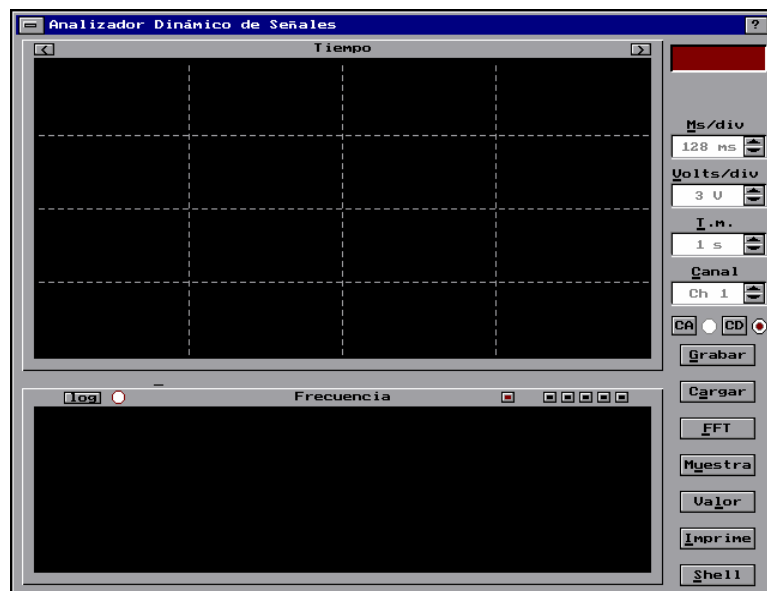


Figura A-2. Interfaz gráfica.

La interfaz gráfica se divide en las siguientes partes:

- La barra de título, la cual se encuentra en la parte superior de la pantalla, y muestra la leyenda Analizador Dinámico de Señales.



- La ventana para la mostrar la señal en forma gráfica en el dominio del tiempo, ésta cuenta con el encabezado "Tiempo" y se localiza en la parte superior izquierda de la pantalla. Cuenta con unas divisiones, el eje X corresponde al tiempo y el eje Y a la amplitud en Volts de la señal; los valores dependerán de la escala en que se encuentren en la ventana de herramientas para control, que se analizará a continuación.



- La ventana para mostrar el espectro de la señal en el dominio de la frecuencia, ésta se encuentra en la parte inferior izquierda de la pantalla y presenta el encabezado de "Frecuencia", sus ejes corresponden: el eje X a la frecuencia en Hertz y el eje Y a la amplitud en unidades.



- La ventana de herramientas para el control de las señales en el dominio del tiempo, ésta se encuentra en la parte superior derecha de la pantalla, su efecto se puede ver controlando alguna señal capturada en memoria por el Analizador Dinámico de Señales. Estas opciones son :



Ms / div: Cambia el número de muestras que se visualizan por cada división de la ventana en el dominio del tiempo. Por omisión, se toma el valor de 128 ms. Los valores que puede tomar son: 4 ,8 ,16 ,32 ,64 , 128, 256, 384, 512, 768, 1024 y 1280. La manipulación de estos valores se analizará mas adelante.

Volts / div : Es la amplitud de la señal, en unidades de Volts por división de la señal que se muestra en la ventana del dominio del tiempo. Como se puede ver esta ventana se divide en el eje Y en cuatro partes iguales, la división media toma el valor de 0, la amplitud positiva se mostrara en la parte positiva del eje y la amplitud negativa en la parte negativa del mismo eje. Por omisión se toma el valor de 3 Volts lo cual indica que la señal puede ser representada si cuenta con un valor máximo de 12 Vpp. El Analizador Dinámico de Señales, puede capturar

señales de hasta 10 Vpp, con lo que todas las señales permitidas se podrán analizar sin dificultad. de igual forma que la opción anterior, los valores que se pueden manejar son 250 mV, 500 mV, 750 mV, 1 V, 1.5 V, 2 V, 2.5 V y 3 V.

T.m. Es el tiempo total de muestreo que queremos visualizar en la pantalla, el Analizador dinámico de señales, captura 5 segundos de la señal, pero con esta opción podemos cambiar el tiempo de muestreo que deseamos analizar. Por omisión se toma el valor de 1 Segundo, los valores que puede tomar son 500 mseg., 1 seg., 1500 mseg., 2 seg., 3 seg., 4 seg. y 5 seg.

Canal. Es la elección del canal que deseamos ver en la ventana del dominio del tiempo. El Analizador Dinámico de Señales cuenta con dos canales de captura de señales. Por omisión, se toma el canal 1, se puede cambiar entre Canal 1, Canal 2 o ambos.

CA y CD. Son los controles para mostrar la señal con sus componentes de Corriente Directa (CD) o de Corriente Alterna (CA). Por omisión se toma la muestra de la señal en CD. Cuando se selecciona alguno de estos controles, se desactiva el otro.

Los controles de < o >. Son los controles para manipular la señal en el dominio del tiempo realizando un movimiento hacia la izquierda o hacia la derecha respectivamente. Solo realizan su función cuando existe una señal almacenada en memoria. El corrimiento será proporcional al valor de la escala de Ms / Div.



La ventana de mensajes, en la cual se visualizan los mensajes de error o de la acción que se éste realizando.



- Los controles para manipular la señal en el dominio de la frecuencia. Estos se encuentran en la parte superior de la ventana de presentación en el dominio de la frecuencia. Estas opciones solo pueden realizar su función cuando existe una señal en memoria y además se ha calculado su espectro, esto se analizará mas adelante.

Log : Cuando se activa este control, la señal en el dominio de la frecuencia, la amplitud de la señal en este dominio se muestra en escala logarítmica, por omisión se muestra en su escala lineal. Este control se activa o desactiva al pulsar el mismo botón.



La ventana de Zoom para el dominio de la frecuencia. Esta ventana aparece en la parte superior derecha de la ventana de presentación en el dominio de la frecuencia. Son 6 Botones, de izquierda a derecha, el primero es el que se activa por omisión y es el que muestra el espectro completo de la señal, es decir, las componentes reales y las complejas de dicho espectro, el segundo, muestra el centro del espectro, también con sus partes real y complejas; el tercero, muestra solo el espectro real de la señal; el cual se divide en tres partes por los tres botones restantes, el cuarto botón muestra un zoom a la primera parte del espectro real, el quinto a la parte del espectro que sigue y por último el sexto muestra la última parte del espectro de la señal, es decir las frecuencias más altas.

- El menú de opciones generales, se encuentra en la parte inferior derecha de la pantalla y es:



Muestra : Esta opción es la que captura por medio de la tarjeta de adquisición de datos la señal a ser analizada. El número total de muestras es de 5120, es decir 5 segundos de muestreo. Estos datos se almacenan en memoria para ser visualizados y manipulados por los demás controles. La muestra de la señal

corresponde al canal que se indica en la opción de Canal. Solo se puede muestrear un canal a la vez, cuando se tiene activada la opción Canal en 1&2 y se activa esta opción, se visualizará un mensaje de "Error".

FFT. Esta opción es la que realiza la transformada rápida de Fourier a la señal que se encuentra localizada en memoria, la cual corresponde al dominio del tiempo. Esta opción calcula el espectro de la señal y devuelve la amplitud de ésta a memoria. Dando como resultado la visualización del espectro en la ventana del dominio de la frecuencia. El cálculo se realiza sobre la señal, ya sea del canal 1 o del 2 respectivamente. Cuando la opción canal se encuentra en 1&2, solo muestra el espectro de ambas señales, cuando anteriormente se ha calculado su espectro por separado.

Valor : Esta opción solo esta disponible con el uso del mouse, al activarse se visualiza una ventana de valores en la parte superior izquierda de la ventana del dominio del tiempo y el cursor toma la forma de una cruceta. Por omisión, el cursor se posiciona en la ventana del dominio del tiempo y se puede posicionar en cualquier punto de la señal, dando como resultado los valores de los ejes, en el tiempo amplitud y número de muestra y en frecuencia amplitud del espectro y número de muestra. Esta opción solo se presenta cuando existe una señal en memoria. Para ver los valores en el dominio de la frecuencia, se debe mantener pulsado el botón izquierdo del mouse y moverlo hacia la posición del espectro, del cual queremos conocer sus valores.

Grabar : Esta opción se utiliza para almacenar la señal al disco, solo almacena la señal en el dominio del tiempo. Se almacena la señal visualizada en el momento de activar esta opción.

Cargar : Esta opción es para cargar de disco a memoria alguna señal almacenada con anterioridad. Se carga en el canal que esta activado.

Imprime : Se utiliza para mandar a imprimir la señal a una impresora gráfica de matriz de puntos. Imprime la ventana en el dominio del tiempo y en el de la frecuencia, además de imprimir los valores representativos de la señal.

Shell : Esta opción es para abandonar temporalmente el software del ADS y regresar al prompt del DOS; tecleando "EXIT" <ENTER>, regresamos al software del ADS.

Como se puede ver, la interfaz gráfica presenta el menú de opciones de una forma fácil de usar, se cuenta con la ayuda de botones para elegir alguna de las opciones a elegir, los cuales se pueden activar por medio del mouse, haciendo un clic sobre el botón a elegir; la otra forma de seleccionar alguna opción es por medio del teclado, esto se hace tecleando la tecla o su combinación para poder acceder al botón o la opción necesaria. La tabla A muestra las diferentes teclas y sus combinaciones para elegir alguna de las opciones disponibles por el programa. Algunas opciones tienen una forma rápida de activarse; éstas son todas las que tienen alguna letra subrayada, por ejemplo Grabar, la cual desde el teclado se puede activar tecleando ALT+G; de igual forma las demás teclas.

Tecla (s)	Acción	Tecla (s)	Acción
ESC	Salir del ADS	M	Cambia las Ms / Div (-)
F1	Menú de ayuda	m	Cambia las Ms / Div (+)
ALT+G	Grabar a disco	V	Cambia los Volts / Div (-)
ALT+A	Cargar de disco	v	Cambia los Volts / Div (+)
ALT+F	Calcula la FFT	C	Cambia el canal (-)
ALT+U	Realiza el muestreo	c	Cambia el canal (+)
ALT+I	Imprime	A	Muestra la señal de CA
ALT+S	Shell	D	Muestra la señal de CD
ALT+O	Escala Logarítmica	1	Muestra espectro completo
ALT+L	Muestra los valores	2,3,4,5,6	Zoom al espectro
'>' o '.'	Hacia derecha (Tiempo)	'<' o ','	Hacia la izquierda (Tiempo)

Tabla A Comandos de activación.

La interfaz gráfica es importante, ya que es la manera con la que el usuario interactúa con el sistema Analizador Dinámico de Señales; es por esto que esta interfaz gráfica se presenta de una manera amigable, proporcionando al usuario la utilización de un mouse para poder elegir el proceso de desee ejecutar dentro de este menú. La elección de algún proceso de este menú, se puede elegir por medio del mouse, al posicionarse en el botón indicado, el cual tiene una leyenda que indica el proceso al que recurre cuando es activado. Otra forma de activar algún proceso es el de usar el teclado, en el listado de procedimientos anterior se muestran las teclas o sus combinaciones

Enseguida, se menciona cada una de las opciones y sus significados en la interfaz gráfica del ADS.

Muestra: Esta opción se utiliza para realizar el muestreo de alguna señal de entrada, el tiempo total de muestreo es de 5 segundos, la señal

muestreada se visualizará en la ventana del dominio del tiempo y sus parámetros serán los que se encuentren en la ventana de control de la señal, es decir, si el valor de la opción canal es 1, la señal muestreada pertenecerá al canal 1, con el valor de Volts por división que tenga dicho control, etc. Esta opción solo está permitida para un solo canal, el 1 o el 2, para visualizar los dos canales, se tendrá que realizar el muestreo por separado. Por ejemplo, muestreando una señal cuadrada de 60 Hz de frecuencia. Lo único que tenemos que hacer es posicionar el mouse sobre el botón "Muestra" o desde el teclado ALT+U, se mostrará en forma de mensaje el número de canal en el cual se está muestreando. La figura A-3 muestra esta acción.

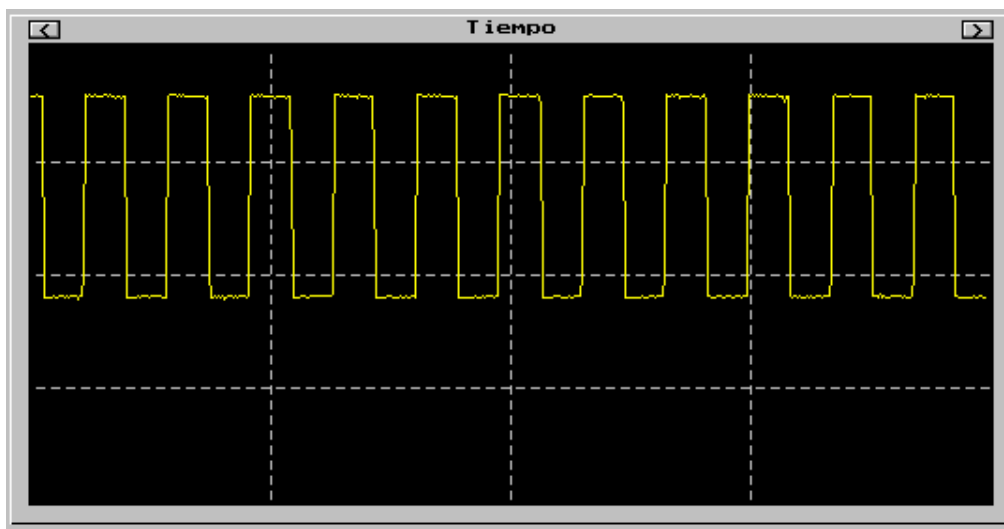


Figura A-3. Muestreo de una señal por el canal 1.

La señal que se capturó, se visualiza con las siguientes propiedades, 128 ms/div, es decir que se visualizan en total 512 muestras, fue muestreado por el canal 1, tiene una amplitud de 11 Vpp, ya que el rango de las divisiones es de 6 Volts / div. y se puede visualizar 1 segundo de muestreo, la parte de la señal que se visualiza es la que se encuentra entre los números de muestra de 513 a 1024.



Si se desea visualizar la parte izquierda de la señal, se posiciona en el botón "<" o se tecldea "<" directamente del teclado, se visualizará la parte de la señal entre las muestras 0 a 512 y es la siguiente:

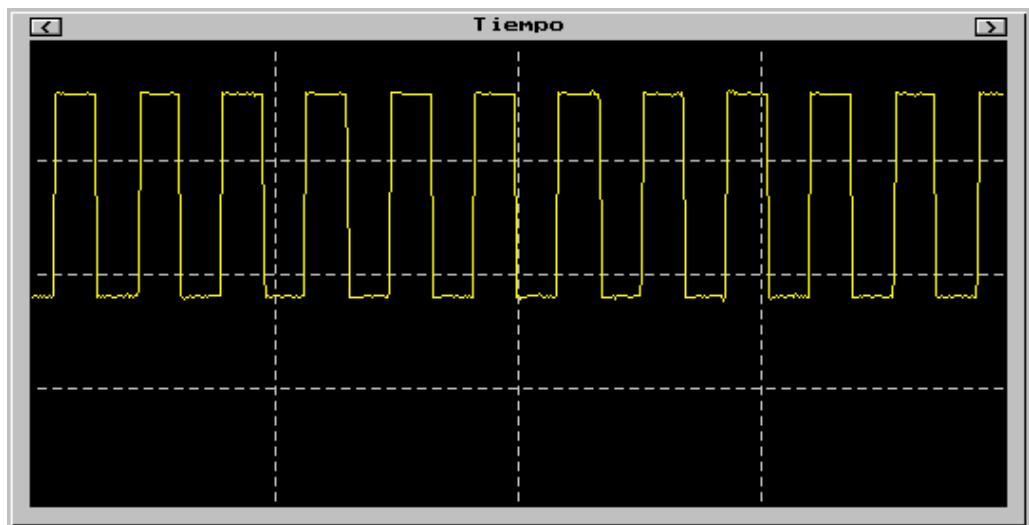


Figura A-4. Ejemplo de utilizar el corrimiento hacia la izquierda.

Si se realiza la acción anterior, aparecerá un mensaje de "Límite" ya que se está visualizando un extremo de la señal, en este caso la que se encuentra entre las muestras de 0 a 512. Para regresar a la señal anterior, es necesario teclear el control ">", de igual forma cuando se llegue a repetir esta misma operación, aparecerá el mismo mensaje indicando el límite de la señal, en este caso el límite es por que solo se desea visualizar 1 segundo de muestreo. Los límites generales son los de las muestras 0 a 5120.

Si esta misma señal se desea visualizar con más detalle, podemos cambiar el rango de voltaje a 3 Volts / div, 64 ms / div y además que se muestren las componentes de corriente alterna; todo esto lo elegimos desde los controles Volts/div, Ms/div, y CA respectivamente.

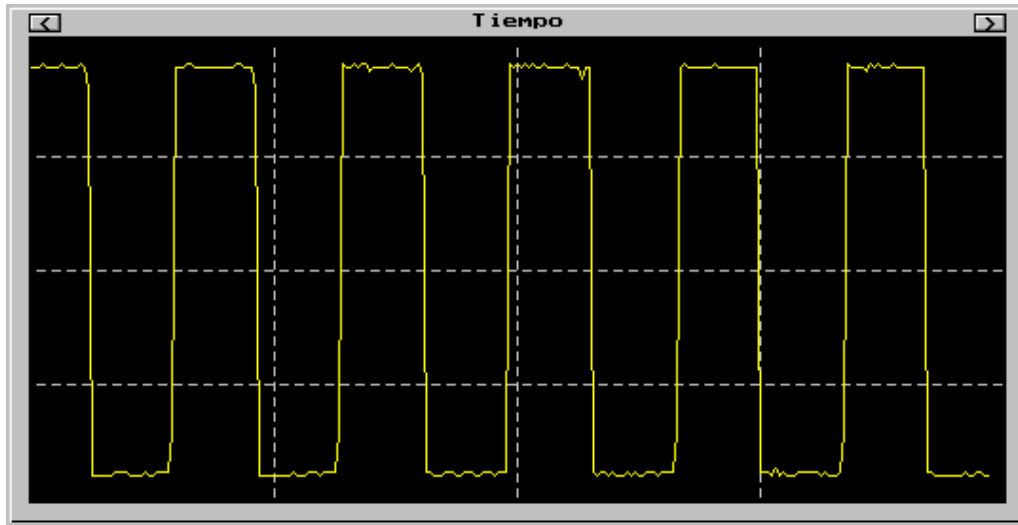


Figura A-5. Ejemplo de modificar las Ms/div.

FFT. Esta opción se utiliza para realizar la Transformada Rápida de Fourier (FFT) a la señal muestreada en el tiempo, se posiciona en el botón FFT o con las teclas ALT+F; si existe señal para realizar la FFT muestra el mensaje de "FFT" de otra forma envía el mensaje de "Sin señal"; Cuando se tienen señales en ambos canales y se visualizan simultáneamente, no se les puede calcular la FFT, se tiene que calcular por separado a cada canal. Por omisión se visualizará el espectro completo de la señal en su escala lineal, de la siguiente forma.

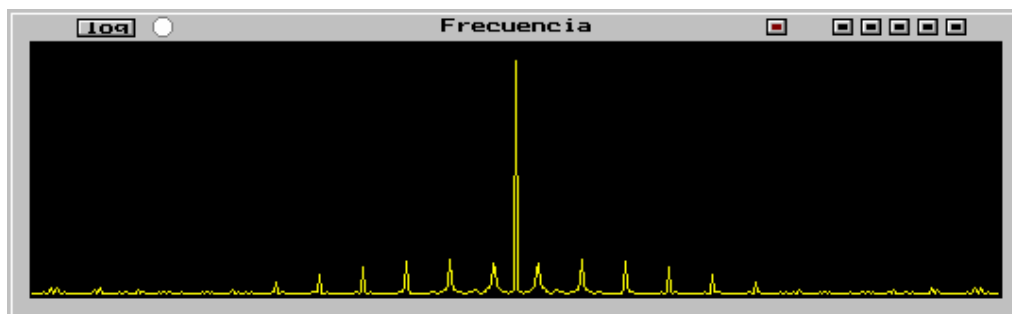


Figura A-6. Visualización del espectro en escala lineal.

Si se desea visualizar el espectro central de la señal, se tecldea el 2o botón del zoom de la ventana del dominio de la frecuencia o simplemente desde el teclado el número 2. El resultado es:

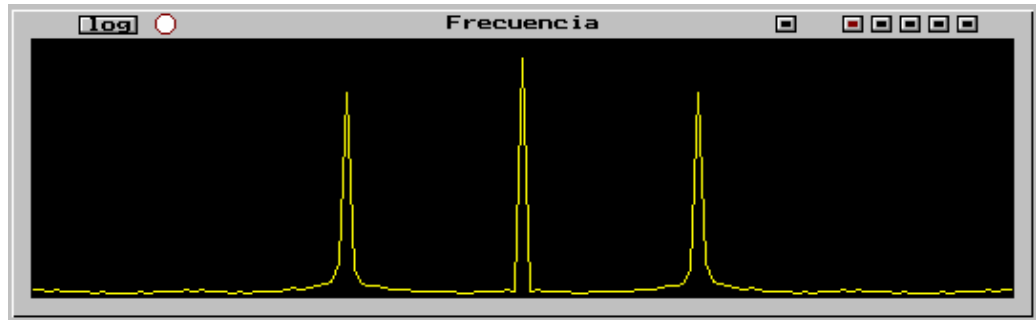


Figura A-7. Visualización del espectro central en escala lineal.

Para poder visualizar el espectro de la señal en escala logarítmica se posiciona con el mouse en el botón Log o tecleando ALT+O, el resultado es:

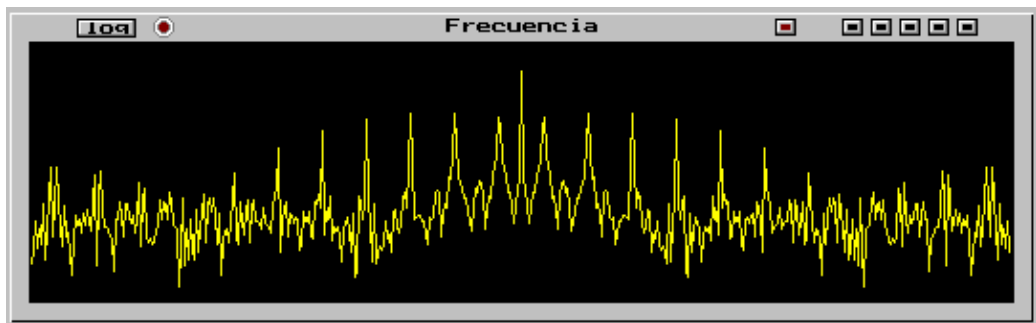


Figura A-8. Visualización del espectro completo en escala logarítmica.

Para visualizar el espectro positivo de la señal se posiciona el mouse en el 3er. botón del zoom de la ventana del dominio de la frecuencia y se hace un clic o simplemente desde el teclado el número 3.

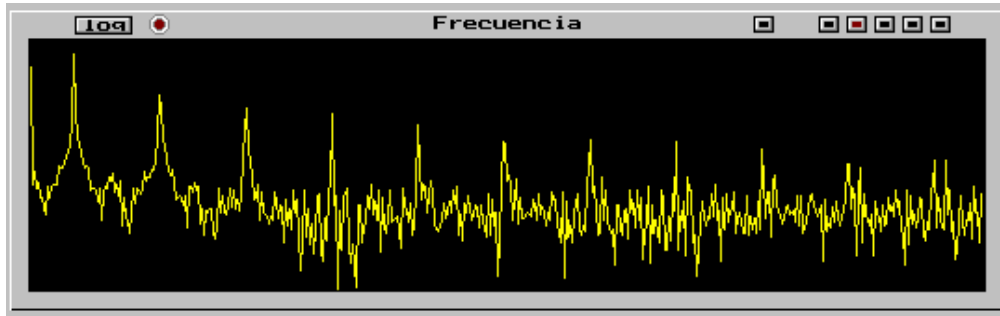


Figura A-9. Visualización del espectro positivo.

Valor: Esta opción se usa para saber por medio de la posición del mouse (esta opción solo esta disponible con el uso del mouse) las amplitudes correspondientes de la señal, por ejemplo, para saber el valor máximo de la señal en el dominio del tiempo, solo se tiene que posicionar el mouse en el botón Valor o tecleando ALT+L, el cursor del mouse tomará la forma de cruzeta y posicionándola en donde se desee, se obtendrán los valores deseados, por ejemplo:

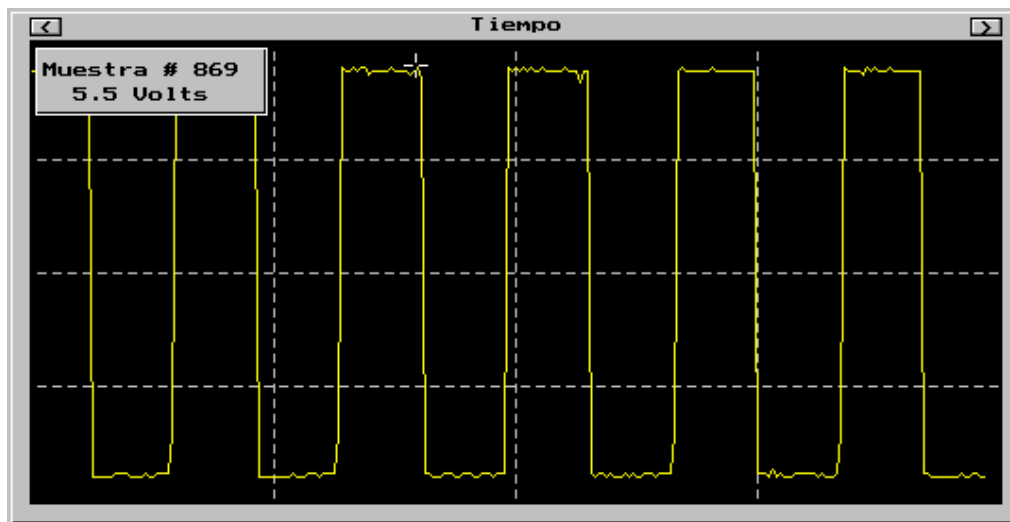


Figura A-10. Ejemplo de utilizar la función Valor en el dominio del tiempo.

lo que nos indica que es la muestra número 869 y su amplitud máxima es de 5.5 Volts. Para obtener los valores en el dominio de la frecuencia, es el mismo procedimiento solo que ahora hay que mantener presionado el botón izquierdo del mouse, el cursor pasará de la ventana del dominio del tiempo al de la frecuencia.

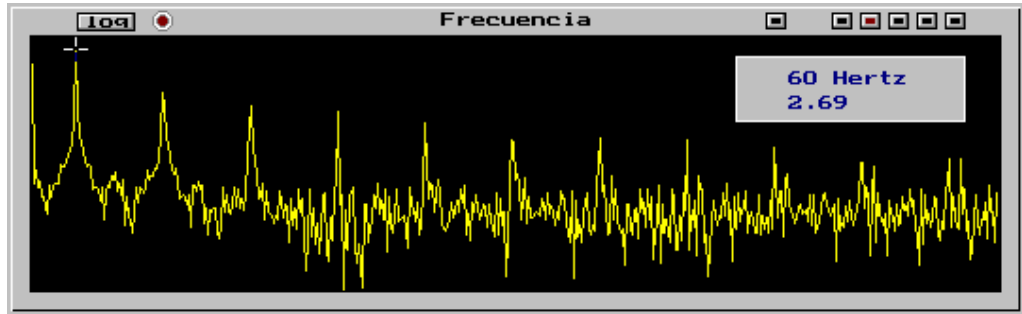


Figura A-11. Ejemplo de utilizar la función Valor en el dominio de la frecuencia.

En esta posición, la amplitud es de 3.69 unidades y la frecuencia es de 60 Hz. Para salir de esta opción solo basta hacer un clic en el botón derecho del mouse o tecleando cualquier tecla desde el teclado.

Grabar: Esta opción es para grabar a disco, se activa posicionándose con el mouse en la tecla Grabar o tecleando ALT+G; aparece una línea con la leyenda "Nombre"; en esta línea se tecla la ruta y el nombre con el que se desea almacenar la señal en el disco; por ejemplo si se quiere grabar con el nombre "CUA60.SIG" aparecerá en la pantalla:

Nombre.. cua60.sig

después de dar un <ENTER>, la señal se grabará en la ruta y con el nombre especificado.

Cargar: Cuando se desea cargar algún archivo de disco, el cual contenga alguna señal almacenada anteriormente por el ADS; se posiciona el mouse en el botón que indica Cargar o desde el teclado ALT+A; la pantalla que aparecerá es la de la figura A-13. En este caso se cargará la señal de disco al ADS en el canal 2, previamente debemos cambiar de canal con "C" o con el mouse en la ventana de control de Canal.



Al activar esta opción aparece el mensaje "Archivo" y una línea con la etiqueta Nombre, debemos teclear el nombre del archivo que contiene la señal que queremos analizar. Después de teclear el nombre se pulsa <ENTER> para realizar la selección, si el archivo no se encuentra en el directorio actual, se visualizará un mensaje de "Error". Si el archivo existe, permanecerá en memoria hasta que carguemos otro archivo o salgamos del ADS. Si existe algún error al teclear el nombre, antes de pulsar <ENTER>, pulsar la tecla BACKSPACE con la que se borrará el nombre incorrecto y se podrá introducir el nuevo nombre corregido.

Nombre . .

Por ejemplo, carguemos el archivo llamado SEN60.SIG, el cual contiene una señal muestreada por el ADS anteriormente. Tecleamos en la línea el nombre de SEN60.SIG y pulsamos <ENTER>. Aparecerá el mensaje de "Cargando", si no encuentra el archivo envía un mensaje de "Error", de otra forma envía el mensaje de "Señal" lo cual indica que se esta cargando el archivo. La figura muestra este resultado.

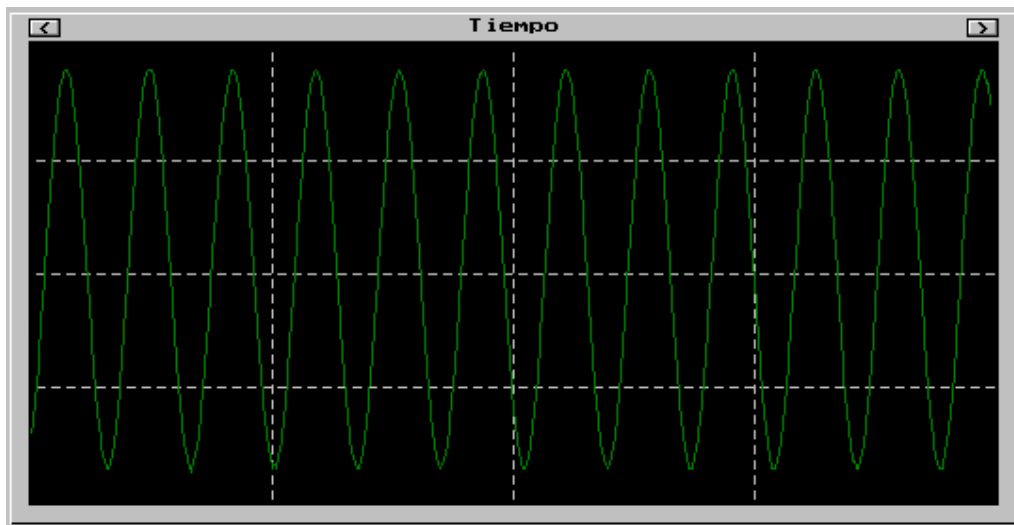


Figura A-14. Función Cargar en el canal 2.

Para que se puedan analizar los dos canales al mismo tiempo, calcular la FFT del canal 2.

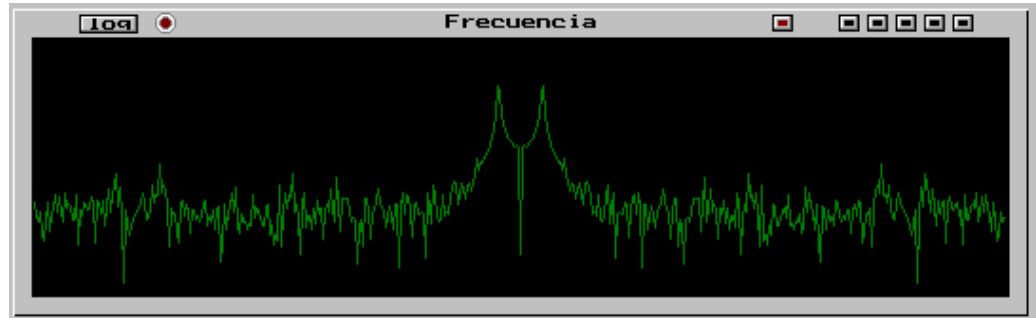


Figura A-15. Cálculo de la FFT en el canal 2.

En esta señal se modificaron las muestras por división a 128 ms/div y 3 Volts/div analizándose las componentes de corriente directa.

Ahora se pueden visualizar los dos canales simultáneamente, cambiando el canal a Canal 1&2, con el control Canal.

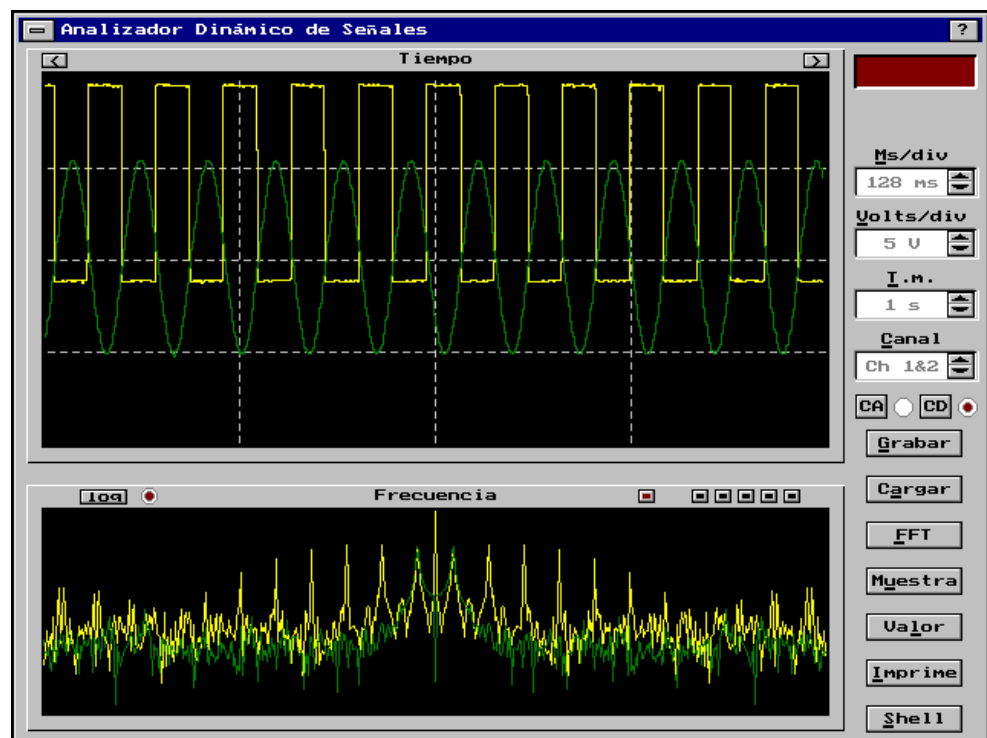


Figura A-16. Visualización simultánea de los canales 1&2.

Como se puede ver, los dos canales se muestran simultáneamente, los controles se cambiaron previamente a 128 Ms/div, 5 Volts/div y Canal 1&2 y se muestran sus componentes de corriente directa. "D".

Imprime: Esta opción utiliza como salida una impresora gráfica de matriz de puntos, se mandan a imprimir las ventanas del dominio del tiempo y el de la frecuencia, además de imprimir los datos con los parámetros importantes como son las Ms/div, Volts/div, Canal y si sus componentes son de corriente directa o corriente alterna. Se activa por medio del botón Imprime o desde el teclado con las teclas ALT+I, cuando se encuentra algún problema con la impresora, se envía un mensaje de error, de otra forma se visualiza el mensaje de "Imprime".

Shell: Esta opción es para abandonar temporalmente el software del ADS sin quitarlo de la memoria de la computadora, se sale al sistema operativo. Se regresa al software del ADS tecleando EXIT + <ENTER>. Esta opción es útil para renombrar archivos, copiarlos u otra tarea que se quiera hacer desde el prompt del sistema operativo. Al activar esta opción lo que se visualizará es:

```

ADS Versión 1.0
Type "EXIT" to return to ADS ...

Microsoft(R) MS-DOS(R) Versión 6.20
(C)Copyright Microsoft Corp 1981-1993.

[C:\ADS>]

```

Figura A-17. Ejemplo de usar la función Shell.

Ayuda : Para poder visualizar un menú de ayuda, en el cual aparecen las teclas más importantes y sus acciones, con el mouse se posiciona en el botón con el signo "?" que se localiza en la parte superior derecha de la pantalla o desde el teclado con la tecla F1. El menú que se muestra es el siguiente:

```

F1      : Menu de Ayuda
ALT+A   : Cargar de Archivo
ALT+G   : Grabar a Archivo
ALT+F   : FFT
ALT+U   : Muestrear
ALT+L   : Mostrar valores
ALT+I   : Imprimir
ALT+S   : Shell
Shift+M : Cambia Ms/div
Shift+U : Cambia Volts/div
Shift+T : Cambia Tiempo de m.
Shift+C : Cambia canal
< ó >  : Recorre en tiempo
1       : Espectro completo
2 .. 6  : Zoom en frecuencia
A       : Muestra CA
D       : Muestra CD
ESC     : Salir de ADS
Version 1.0

```

Figura A-18. Visualización del menú de ayuda.