

92

UNIVERSIDAD NACIONAL AUTONOMA
DE MEXICO



FACULTAD DE INGENIERIA

SISTEMA AUTOMATIZADO PARA POSICIONAR
EN UN PLANO X-Y, UNA SONDA DE INTENSIDAD
SONORA

280067

T E S I S

QUE PARA OBTENER EL TITULO DE:
INGENIERO MECANICO ELECTRICISTA
(AREA: ELECTRICA Y ELECTRONICA)

P R E S E N T A :
ABEL VILLANUEVA PEÑA

DIRECTOR DE TESIS: M.C. SANTIAGO JESUS PEREZ RUIZ



CIUDAD UNIVERSITARIA, MEXICO, D.F.

2000



Universidad Nacional
Autónoma de México



UNAM – Dirección General de Bibliotecas
Tesis Digitales
Restricciones de uso

DERECHOS RESERVADOS ©
PROHIBIDA SU REPRODUCCIÓN TOTAL O PARCIAL

Todo el material contenido en esta tesis esta protegido por la Ley Federal del Derecho de Autor (LFDA) de los Estados Unidos Mexicanos (México).

El uso de imágenes, fragmentos de videos, y demás material que sea objeto de protección de los derechos de autor, será exclusivamente para fines educativos e informativos y deberá citar la fuente donde la obtuvo mencionando el autor o autores. Cualquier uso distinto como el lucro, reproducción, edición o modificación, será perseguido y sancionado por el respectivo titular de los Derechos de Autor.

INDICE

RECONOCIMIENTOS Y DEDICATORIAS	I
PROLOGO	II
CAPITULO 1	
1. INTRODUCCION	1
1.2 DESCRIPCION DEL SISTEMA	3
CAPITULO 2	
2. DEFINICION DE INTENSIDAD SONORA	7
2.1 METODOS DE MEDICION DE INTENSIDAD SONORA	17
2.2 APLICACIONES	21
2.3 METODO DE MEDICION EMPLEADO	23
REFERENCIAS CAPITULO 2	27
CAPITULO 3	
3. SISTEMA DE POSICIONAMIENTO Y ORIENTACION DE LA SONDA	28
3.1 SISTEMA MECANICO	29
3.2 POSICION	34
3.2.1 MOTORES DE PASOS	34
3.2.2 MOTORES DC	36
3.2.3 CIRCUITOS DE CONTROL PARA LOS MOTORES	37
3.3 ORIENTACION	42
3.3.1 METODO PARA SENSAR LA POSICION	43
3.3.2 SENSADO DE LA POSICION	47
3.4 PROGRAMAS	55
REFERENCIAS CAPITULO 3	58
CAPITULO 4	
4. SISTEMA DE ADQUISICION Y DE PROCESAMIENTO DE DATOS	59
4.1 TARJETA DE INTERFAZ	60
4.2 ANALIZADOR DE FRECUENCIAS 2133 DE B&K	60
4.3 SONDA DE INTENSIDAD SONORA TIPO 3545	62
4.4 PROGRAMA DE CONTROL WT9378	63
4.4.1 COMUNICACION ENTRE EL ROBOT Y EL WT9378	64
4.4.2 INTEGRACION Y FUNCIONAMIENTO DE LOS SISTEMAS	67
REFERENCIAS CAPITULO 4	69
CAPITULO 5	
5. PRUEBAS DEL SISTEMA	70
5.1 CALIBRACION	72
5.2 PRUEBA DE CARACTERIZACION DE LA GUITARRA ACUSTICA CLASICA	75
REFERENCIAS CAPITULO 5	86
CAPITULO 6	
6. CONCLUSIONES	87
APENDICE A	88
APENDICE B	95
APENDICE C	105
APENDICE D	116
BIBLIOGRAFIA	119

RECONOCIMIENTOS Y DEDICATORIAS

A mis padres, por ser el origen de lo que soy, por su paciencia, comprensión y confianza en que pueda lograr mis metas.

A mis hermanas, por toda su ayuda y por que son totalmente parte de mis logros.

A mis hermanos, por que de hecho sé que pueden superar lo que hasta hoy he logrado.

A Marisol, por su apoyo incondicional en todo el tiempo que la conozco y por ser la persona que más fe y paciencia me tiene.

A mis sobrinas, con quienes he pasado muchos momentos bonitos y además de que les deseo todo lo mejor.

Al M.C Santiago J. Pérez Ruiz. Por su apoyo, paciencia y su amistad. También a su esposa Rosalba Castañeda y a Chema.

A los cuates, que ojalá y algún día comprenda él por que nunca tenia tiempo para estar con ellos.

Al Centro de Instrumentos y a los cuates de acústica (Toño, Oscar P, Oscar C, Marcos, Ricardo, Raúl, etc.)

Y a todos las personas que de alguna forma han contribuido a lo que soy.

PROLOGO

Bajo la condicionante de diseñar un sistema que pudiese colocar a una sonda de intensidad sonora en un plano $X-Y$, para que al final se tuviera un sistema automático controlado con una computadora personal, para la elaboración de mapas de intensidad sonora (efectuando un barrido en forma discreta) en la sección de acústica del Centro de Instrumentos de la UNAM. Comenzó el reto que represento este proyecto.

En un principio y con los conocimientos adquiridos en la facultad y con información adicional, diseñe el control (electrónico y los programas) de los motores que moverían las partes mecánicas que se encargarían de colocar la sonda en diversos puntos de un plano $X-Y$. Una vez diseñado esto fue necesaria la construcción mecánica, del que yo he dado el nombre de 'robot'. Posteriormente se eligió el método para conocer la posición de la sonda que estaría montada sobre la cabeza de este robot. Enseguida fue necesario incluir en los programas de control del robot algunas rutinas de error, previendo alguna falla del mismo. Lo anterior significo una gran inversión en tiempo, tanto para documentación como para implementación. Pero no termina aquí este asunto ya que una vez concluida esta actividad, tuvo que ser necesario, el conocer a los demás equipos de medición a utilizar en el proyecto, así como ver la forma de utilización para que estos, junto con el robot me entregaran al final una gráfica ó un mapa de intensidad sonora. Esta actividad me obligo al estudio teórico de la cantidad física conocida como intensidad sonora, posteriormente realice pruebas de programación de los instrumentos de medición con la interfaz IEEE-488 (quien también requirió su estudio para su buena utilización) el resultado de estos programas de prueba fue el conocer la manera de programación de equipos de medición sofisticados, controlados con una computadora personal, después de las pruebas se utilizo un programa comercial que pudo controlar los equipos de medición y que en principio podría controlar también un robot, esto a primera vista significaba una gran inversión de tiempo para encontrar la forma de incluir los programas controladores del robot (compilados en Microsoft C versión 7.0), dentro del programa comercial, me preguntaba en aquel entonces que si esto era posible tendría solucionado el problema que representaba el proyecto de tesis aquí descrito. El porque de la afirmación anterior, obedecía a que ya tenia por un lado un robot que podía mover su cabeza a placer y por lo tanto colocar de esta forma la sonda de intensidad sonora en un punto previamente definido, y por el otro lado tenia un programa de computadora que controlaba a todos los instrumentos de medición que esta diseñado específicamente para obtener mapas de intensidad sonora y que a pesar de ser un programa comercial permitía la facilidad de incrustar rutinas para el control de robots por medio de archivos de inicio y arranque, y de un programa hecho en pascal. Afortunadamente fue posible incluir dentro de este programa en pascal mis programas hechos en C.

Al poder hacer lo anterior pude conformar un sistema ó un equipo automático de mapeo de intensidad sonora, que estaba bajo prueba de operación. En estas pruebas se

logro saber que el tiempo máximo para un barrido de 192 puntos (12 en el eje X por 16 en el Y), se hacían en un tiempo máximo aproximado de 30 min, tiempo en el que el operador puede efectuar otras actividades. Después se hicieron varios programas en el paquete Matlab que dan un postprocesamiento a la información dada por el programa comercial de intensidad sonora (WT9378 nombre comercial). Para este postprocesamiento fue necesario encontrar la manera de abrir archivos generados por el programa WT9378, que contenían la información útil para el postprocesamiento, se intentó abrirlos con el C, a final de cuentas se abrieron con programas en Matlab, la información de estos archivos se volvió a analizar pero ahora con todas las ventajas que permite este paquete.

El resultado final fueron mapas de intensidad sonora, con diferentes formatos de presentación, mayor resolución y mayor facilidad para el análisis de resultados.

Finalmente se probó el sistema elaborando los mapas de intensidad en una fuente sonora compleja representada por una guitarra clásica.

El resultado de este trabajo es el siguiente informe el cual pongo a consideración de los lectores esperando que sea de su utilidad.

CAPITULO 1

I. INTRODUCCION

Del sistema automatizado para posicionar en un plano $X-Y$, una sonda de intensidad sonora. Se presenta a continuación su

OBJETIVO:

Contar con un sistema para posicionar una sonda de intensidad sonora (Brüel & Kjør Mod. 3545) en un plano $X-Y$, para poder hacer mapas de intensidad de fuentes acústicas.

Este proyecto surge de una de las necesidades que se tienen en la sección de acústica en el Centro de Instrumentos de la UNAM; la de contar con un sistema que permita colocar una sonda de intensidad sonora en las coordenadas (x,y) , además de adquirir y procesar datos de presión sonora, a fin de poder elaborar mapas de intensidad sonora de diversas fuentes acústicas de manera automática utilizando para esto una computadora personal.

Para comprobar este sistema se propuso evaluar, como fuente sonora compleja, a la guitarra clásica. Debido a que este hecho representa un apoyo importante en las investigaciones sobre la guitarra y otros instrumentos musicales, que se realizan dentro de la sección de acústica.

Hasta el momento en que se inició el desarrollo de este sistema, dentro de la sección de acústica no se disponía de un medio automático que pudiese hacer este tipo de mapeos. La manera en que se hacía, era colocar la sonda manualmente entre los puntos de unión de una rejilla hecha de hilos o alambres, la cual representaba un plano $X-Y$, en cuanto a la adquisición de los datos provenientes de la sonda, se venían haciendo de forma semiautomática, utilizando una computadora personal (PC) y equipo adicional de instrumentación, la desventaja de este procedimiento es que para elaborar un mapa de intensidad sonora de cualquier tipo fuente acústica era necesario que cuando menos dos personas participaran en dicho proceso, una para colocar la sonda en el punto de medición y la otra para manejar los instrumentos de medición. En la figura 1.1 se puede ver como se hacían este tipo de mapas

Para justificar el desarrollo de este sistema basta puntualizar, que la elaboración de dichos mapas consumía mucho tiempo por lo que el automatizar este proceso ahora permite un menor consumo de tiempo para el operador, además de mayor flexibilidad y precisión en este tipo de mediciones. Actualmente existen en el mercado (importación) sistemas similares, pero estos sistemas tienen el inconveniente, además de su excesivo

precio, de no contar con la flexibilidad necesaria que demanda el trabajo de investigación.

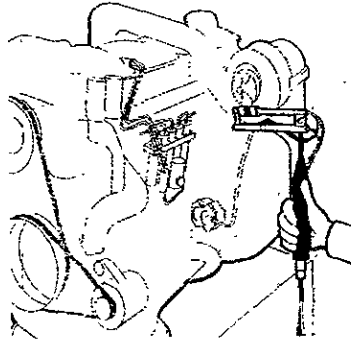


Figura 1.1 Nótese como el usuario tiene que colocar la sonda, en forma manual en el punto de unión de la rejilla.

El sistema elaborado, cuenta con mayores ventajas, puesto que su desarrollo fue de un bajo costo monetario, además cuenta con la suficiente versatilidad como para utilizarse en investigación.

La automatización del proceso para la elaboración de los mapas de intensidad sonora, permite contar con una técnica de medición poderosa, que revela aspectos de los campos sonoros que no son obvios por otros métodos de medición. Al medir intensidad sonora, se está obteniendo una cantidad vectorial que describe la magnitud y dirección del flujo de energía acústica que pasa a través de una área unitaria. En cambio cuando se mide la presión sonora con los métodos tradicionales, se está midiendo el nivel total de la presión acústica que se capta por un micrófono, incluyendo en esta medida, todas las contribuciones sonoras, de todas las fuentes acústicas radiando en ese momento. Entre otras características, la intensidad sonora es sensible únicamente al flujo neto de energía sonora, de ahí que cualquier efecto debido a la parte reactiva del campo sonoro (tal como ondas estacionarias) y efectos producidos por otras fuentes acústicas, se rechace en la medición, por lo que con esta técnica es posible saber la contribución sonora individual, de diversas fuentes radiando al mismo tiempo, evitando con esto el trasladar la fuente sonora a un laboratorio acústico especializado en donde se puede aislar de otras fuentes de ruido.

Adicionalmente, la medición de intensidad sonora no sólo nos proporciona información sobre la cantidad de flujo neto de energía sonora, sino que además nos proporciona información sobre la dirección de este flujo, pudiendo, de esta manera, localizar, zonas de máxima (o mínima) emisión de energía; facilitando la tarea de localizar e identificar fuentes sonoras. Al medir intensidad sonora en varios puntos localizados dentro de una cierta área y mediante un procesamiento adicional, se puede

construir un mapa de intensidad sonora de la fuente acústica, que nos puede revelar el comportamiento de esta. Ver figura 1.1.

Para formar el mapa es necesario hacer que la sonda haga un barrido cubriendo la superficie de interés, el barrido puede ser en forma continua o como en nuestro caso en forma discreta. Entonces para llevar a cabo el barrido sobre la superficie, se diseñó y construyó un sistema que puede posicionar, en forma automática, a la sonda de intensidad sonora en diversos puntos dentro del área bajo estudio.

1.2 DESCRIPCION DEL SISTEMA

El sistema que se desarrolló consta de dos etapas, una es la etapa de posicionamiento y otra que consiste en el desarrollo de un programa para enlazar al programa que efectúa la adquisición y procesamiento de los datos.

La etapa de posicionamiento esta compuesta por un sistema mecánico de 2 coordenadas construido con materiales de bajo costo, el sistema tiene dos grados de libertad (x,y) y permite colocar, una sonda de intensidad sonora, en un máximo de 192 posiciones (barrido en forma discreta). El sistema se controla mediante una computadora personal a través de su puerto paralelo y tres tarjetas de expansión del puerto, además de sus respectivos programas de control, a este sistema le llamaremos robot. En el capítulo 3 se mencionan los detalles de su diseño y construcción, los detalles de la sonda y del barrido se abordan en los capítulos 4 y 2 respectivamente.

La adquisición y el procesamiento de los datos adquiridos se hacen con los siguientes elementos: una sonda de intensidad acústica, una unidad de control remoto, un analizador de frecuencias, una computadora personal y un programa adecuado para el análisis de los datos de intensidad sonora. En esta parte se hacen las mediciones y el despliegue de los resultados de intensidad sonora.

El sistema que automatiza este proceso esta formado por: la sonda de intensidad sonora Brüel & Kjær Mod. 3545. El analizador de frecuencias Brüel & Kjær Mod. 2133, una computadora personal, la interfaz para instrumentos de medición conocida como GPIB-488 ó IEEE-488, un robot (sistema mecánico) con sus programas incluyendo sus tarjetas electrónicas de control y el programa comercial WT9378 para el procesamiento de los datos acústicos (también de la marca Brüel & Kjær) el cual sirve de enlace entre el robot y los instrumentos de medición por medio de la computadora.

Los detalles de la sonda, del analizador, del programa WT9378, del bus de propósito general cuyo protocolo es el IEEE-488, y la tarjeta NI-488, se abordan en el capítulo 4.

Al unir la etapa de posicionamiento con la etapa de adquisición y análisis de datos, queda conformado el sistema de automatización, para la elaboración de mapas de intensidad acústica. En la figura 1.2 se puede ver un esquema del sistema desarrollado.

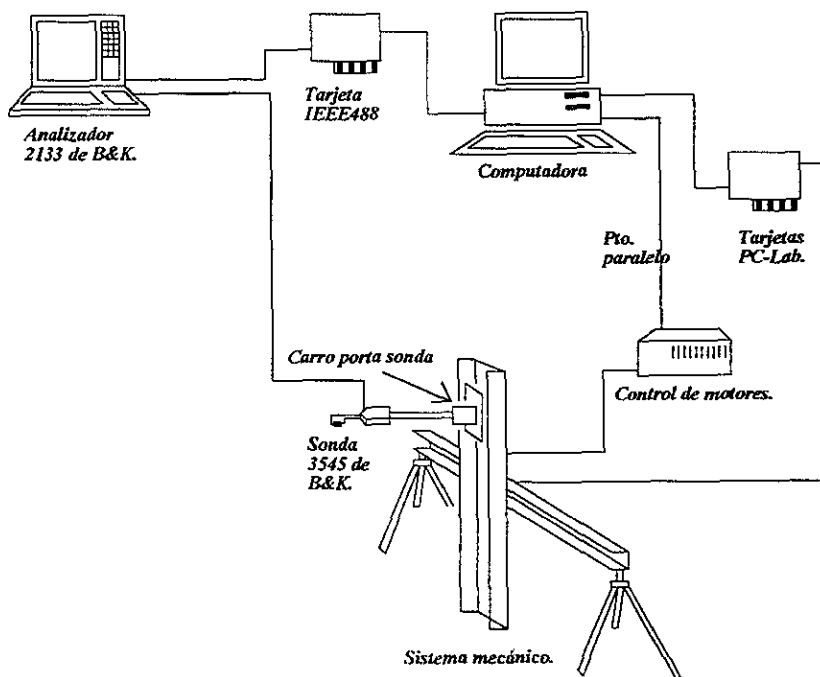
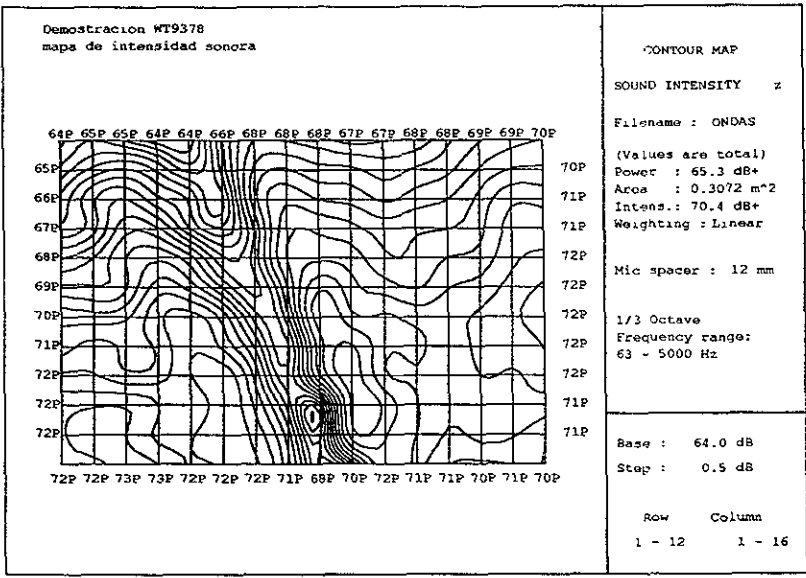


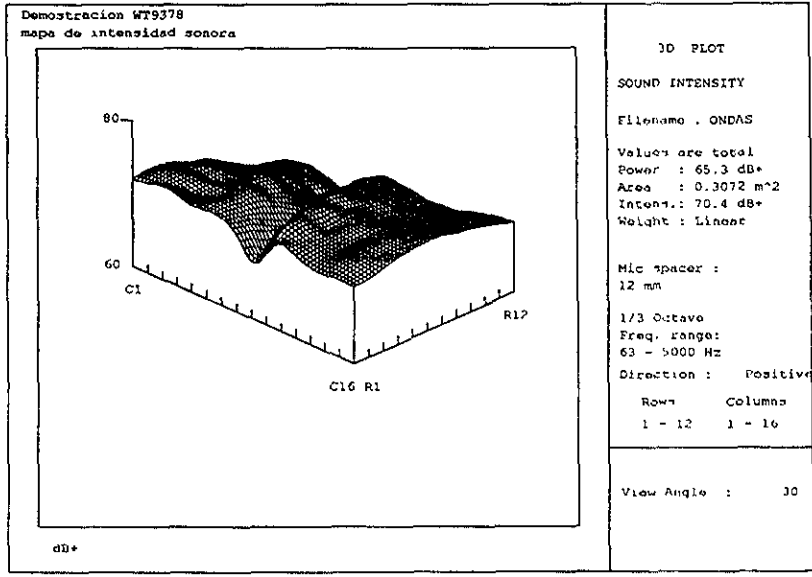
Figura 1.2 Esquema del sistema de automatización, para la elaboración de mapas de intensidad sonora, entiéndase por *sistema mecánico*, al sistema de posicionamiento y orientación de la sonda en el plano $X-Y$ (robot).

Como muestra de los resultados obtenidos en el capítulo 5 es posible ver los mapas de intensidad sonora que caracterizan a una guitarra clásica. En esta introducción se incluyen algunos de los diferentes formatos para los mapas de intensidad sonora, que el sistema puede elaborar.

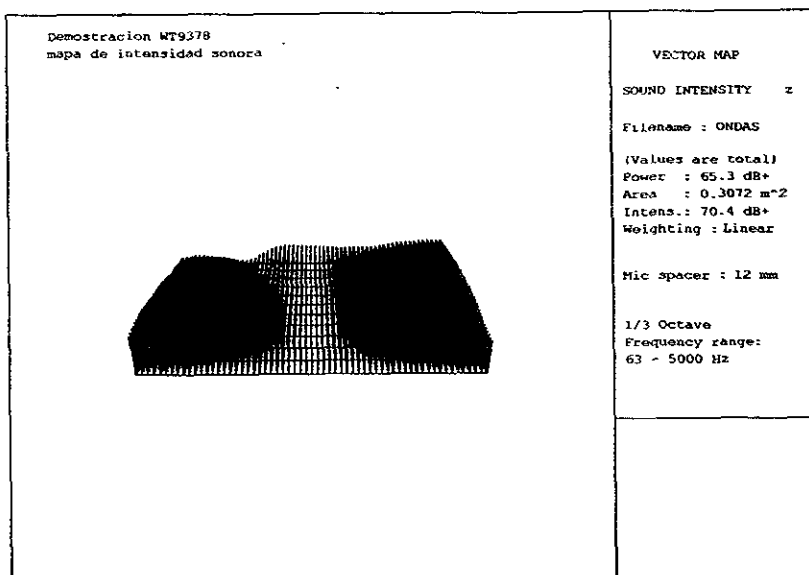
Los mapas aquí presentados son los que se obtuvieron en una de las pruebas realizadas y que se explica en el capítulo 5. Los mapas no están hechos en el programa de postprocesamiento (realizado en Matlab), pero sí con el programa WT9378.



Mapa 1 Mapa de contorno para intensidad sonora, hecho en el WT9378.



Mapa 2 Mapa en 3D de intensidad sonora, para las mismas mediciones del mapa 1, hecho en el WT9378



Mapa 3 Mapa vectorial de intensidad sonora, para las mismas mediciones de los mapas 1 y 2, hecho en el WT9378.

Las aplicaciones que puede tener este sistema pueden ser:

- Caracterización de instrumentos musicales.
- Control de ruido (en salas de conciertos, edificios, barcos, aviones, automóviles, fábricas, etcétera).
- Caracterización del campo sonoro de cualquier fuente de radiación sonora compleja.
- Identificación de zonas de radiación o absorción de energía.

CAPITULO 2

2. DEFINICION DE INTENSIDAD SONORA

Las ondas acústicas que producen la sensación de sonido (también hay ondas ultrasónicas e infrasónicas cuyas frecuencias están fuera de los límites audibles), son parte de una variedad de perturbaciones de presión que se pueden propagar a través de un medio compresible (líquidos y gases), las perturbaciones, están asociadas con los pequeñísimos movimientos vibratorios (periódicos y no periódicos) del medio. Las vibraciones sonoras son del tipo longitudinal, las partículas del medio, se mueven de un lado a otro en la dirección de propagación de la onda, produciendo regiones adyacentes de compresión y rarefacción. Como resultado, el cambio de presión, que ocurre cuando un fluido se expande o se comprime, es la única fuerza restauradora capaz de propagar una onda.

La propagación del sonido siempre esta asociado con algún medio, el medio fluido al que se suscribe este trabajo es el gaseoso, en concreto el aire; las ondas sonoras, no se propagan en el vacío. El desplazamiento relativo de las partículas, en la propagación de la onda sonora, en el aire, involucran cambios locales (generalmente pequeños) en la presión, densidad, temperatura del medio y la velocidad de las partículas. Para relacionar estas variables en el estudio de este fenómeno se emplea la ecuación de Euler (que es la segunda ley de Newton aplicada a la mecánica de fluidos), la ley de los gases y la ley de conservación de masa. También hay que considerar que el fenómeno es de tipo adiabático, dado que el termino partícula del fluido significa un elemento de volumen lo suficientemente grande para contener millones de moléculas de aire y pensar en el fluido como un medio continuo, y sin embargo tan pequeño que se pueda considerar que todas las variables acústicas son casi constantes en todo el elemento de volumen. Dado que las moléculas de un fluido no tienen posiciones medias fijas; aun sin la presencia de una onda, están en constante movimiento, con velocidades promedio mucho mayores que cualquier velocidad de partícula asociada con el movimiento ondulatorio. Sin embargo, se puede tratar a un pequeño volumen como una unidad permanente sin cambio, ya que aquellas moléculas que dejan sus confines son reemplazadas por un número de moléculas igual que poseen (en promedio) propiedades idénticas, de tal manera que las propiedades microscópicas permanecen sin cambio, y poder considerar a las ondas sonoras como un proceso adiabático

De la ecuación de estado adiabática para un gas perfecto se deriva la siguiente ecuación, que relaciona la presión acústica con la densidad del medio (aire)¹.

$$\frac{\partial p}{\partial t} = c^2 \frac{\partial \rho}{\partial t} \quad (2.1)$$

donde

p presión acústica en [pa]

ρ densidad del medio en $[\text{kg}/\text{m}^3]$
 c^2 velocidad de propagación del sonido $[\text{m}/\text{s}]$

En el caso de la propagación de sonido en forma unidimensional se tienen las siguientes ecuaciones:

$$p = c^2 \rho \quad (2.2)$$

De la segunda ley de Newton se deriva la ecuación que relaciona a la presión densidad y velocidad de partícula. Que en la mecánica de fluidos es conocida como ecuación de Euler².

$$-\nabla p = \rho_0 \frac{\partial \vec{u}}{\partial t} \quad (2.3)$$

donde

ρ_0 es la densidad de equilibrio constante del medio (fluido)
 \vec{u} vector velocidad de partícula $[\text{m}/\text{s}]$

En forma unidimensional:

$$\frac{\partial u_x}{\partial t} = -\frac{1}{\rho_0} \frac{\partial p}{\partial t} \quad (2.4)$$

Para relacionar la velocidad de partícula con la densidad se llega a la ecuación de continuidad linealizada, que nos relaciona al fluido con su compresión o dilatación.

$$\nabla \vec{u} = -\frac{1}{\rho_0 c^2} \frac{\partial p}{\partial t} \quad (2.5)$$

En forma unidimensional.

$$\frac{\partial u_x}{\partial x} = -\frac{1}{\rho_0 c^2} \frac{\partial p}{\partial t} \quad (2.6)$$

Dichas leyes relacionan a las variables, en una ecuación conocida como ecuación de onda. Su forma depende de la naturaleza del movimiento de onda, de la naturaleza de la onda de presión en el medio y del sistema de coordenadas empleado.

Así que para el caso de la propagación del sonido en forma unidimensional (a lo largo del eje x) se tiene la siguiente ecuación de onda:

$$\frac{\partial^2 p}{\partial x^2} + \frac{\partial^2 p}{\partial y^2} + \frac{\partial^2 p}{\partial z^2} - \frac{(1/c^2) \partial^2 p}{\partial t^2} = 0 \quad (2.7)$$

Para el caso en tres dimensiones, la ecuación de onda más común es la que esta en términos de la presión acústica.²

$$\frac{\partial^2 p}{\partial x^2} + \frac{\partial^2 p}{\partial y^2} + \frac{\partial^2 p}{\partial z^2} - \frac{(1/c^2) \partial^2 p}{\partial t^2} = 0 \quad (2.8)$$

Las soluciones de la ecuación de onda, para los casos de una onda plana y una onda esférica se mencionan a continuación. En la onda plana, las variables acústicas del campo sonoro son uniformes y se conoce como " campo de onda plana ". A si que para la ecuación de onda reducida a una forma unidimensional , ecuación (2.7), su solución general esta representada por dos ondas progresivas, de la siguiente forma³

$$p(x,t) = f(ct - x) + g(ct + x) \quad (2.9)$$

donde

f, g son funciones que dependen del espacio y el tiempo.

La función f es una onda que viaja en el sentido positivo del eje x , la función g es una onda que viaja en el sentido negativo del eje x .

La forma compleja de la solución armónica para la presión acústica de una onda plana que se propaga en un solo sentido es³:

$$p = Ae^{j(\omega t - kx)} + Be^{j(\omega t + kx)} \quad (2.10)$$

donde

- A y B son las amplitudes de presión complejas.
- ω frecuencia de la onda.
- k numero de onda.
- x dirección de propagación.

Aplicando (2.3) a (2.10), se logra obtener la distribución de la velocidad de partícula asociada

$$u(x,t) = \left(\frac{1}{\rho_0 c}\right) A e^{j(\omega t - kx)} - \left(\frac{1}{\rho_0 c}\right) B e^{j(\omega t + kx)} \quad (2.11)$$

donde

$\rho_0 c$ es conocida como la impedancia acústica en el fluido [rayls].

De (2.10) y (2.11) obtenemos las velocidades de partícula en ambos sentidos¹:

$$u_+ = + \frac{p_+}{\rho_0 c} \quad \text{y} \quad u_- = - \frac{p_-}{\rho_0 c} \quad (2.12)$$

Una ecuación que puede ser más útil es la ecuación de onda esférica, para campos de presión esféricamente simétricos, se expresa de la siguiente forma¹:

$$\frac{\partial^2(pr)}{\partial r^2} = \frac{1}{c^2} \frac{\partial^2(pr)}{\partial t^2} \quad (2.13)$$

donde

pr es la variable dependiente del producto entre presión y la distancia radial.

Entonces la solución general a la ecuación de onda esférica es:

$$p(r,t) = \frac{1}{r} [f(ct-r) + g(ct+r)] \quad (2.14)$$

Las ondas divergentes más importantes son armónicas, por lo que se puede representar a la presión en forma compleja³.

$$p(r,t) = \frac{A}{r} e^{j(\omega t - kr)} + \frac{B}{r} e^{j(\omega t + kr)} \quad (2.15)$$

donde

- p es la onda de presión compleja
- A es la amplitud de la presión compleja.
- k es el número de onda acústica, quien representa al cambio de fase con la distancia, ω/c .

ω es la frecuencia angular, que representa el cambio de fase en el tiempo.
 r distancia radial.

Aplicando (2.3) a (2.15) se obtiene la velocidad de partícula radial.

$$u_r(r, t) = \left(\frac{p(r, t)}{\rho_0 c} \right) \left(1 - \frac{j}{kr} \right) \quad (2.16)$$

Hasta aquí se ha puesto énfasis en la presión y la velocidad de partícula, por la siguiente razón; cuando se propaga la onda sonora, a causa del fenómeno acústico, se generan regiones de dilatación y compresión de las partículas, cuando esto ocurre, se esta liberando energía, la cual se reparte uniformemente sobre las regiones de expansión y compresión. Estas ondas se propagan a partir de la fuente en dirección perpendicular al eje de propagación, y su superficie se incrementa a medida que se alejan de dicha fuente.

La concentración de energía por unidad de superficie de onda disminuye de acuerdo se aleja de la fuente. Esta concentración de energía, es llamada *intensidad sonora instantánea* " I " (cantidad vectorial). En otras palabras es la cantidad de energía que fluye de una fuente sonora (representada por un punto), y que pasa a través de una área unitaria, en dirección perpendicular a dicha área⁴. Ver figura 2.1

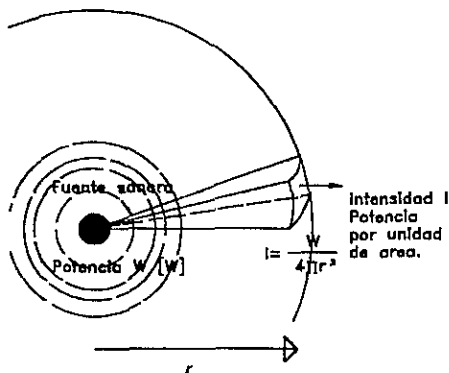


Figura 2.1 Representación de la intensidad sonora, que pasa por una superficie esférica con área igual a $4\pi r^2$

Dado que la intensidad sonora es energía por área, es fácil ver que también es igual a la presión multiplicada por la velocidad (de ahí el énfasis puesto en estas dos variables).

$$\begin{aligned} \text{Intensidad} &= \frac{\text{Potencia}}{\text{Area}} = \frac{\text{Energía}}{\text{Area} * \text{Tiempo}} \\ &= \frac{\text{Fuerza} * \text{Distancia}}{\text{Area} * \text{Tiempo}} = \text{Presión} * \text{Velocidad} \left[\frac{\text{W}}{\text{m}^2} \right] \end{aligned}$$

Entonces la intensidad sonora instantánea " I " es:

$$I = p u \left[\frac{\text{W}}{\text{m}^2} \right] \quad (2.17)$$

Donde

- u es el vector velocidad de partícula instantánea
- p es la presión sonora instantánea.

En la mayoría de los textos se indica la *Intensidad sonora " I "*, en lugar de la *intensidad sonora instantánea " I "*, debido a que la primer cantidad es la más extensamente medida, por que indica el flujo neto de energía que libera la fuente. La *Intensidad sonora " I "* se define como la rapidez "promedio" del flujo de energía a través de una área unitaria normal, en la dirección de propagación de la onda. Sus unidades fundamentales son watts por metro cuadrado (W/m^2), La rapidez instantánea con que un elemento del fluido hace trabajo por unidad de área en un elemento adyacente es pu ⁵. La intensidad es el promedio temporal de esta rapidez.

$$I = \langle pu \rangle = \frac{1}{t} \int_0^t pu \, dt \quad (2.18)$$

donde

- $\langle \rangle$ indica la promediación en el tiempo.
- I intensidad sonora.
- p presión sonora instantánea.
- u velocidad de partícula instantánea.

Aplicando (2.17), a un campo de onda plana, dado por las ecuaciones (2.10) y (2.11), encontramos que la intensidad sonora es:

$$I = \frac{p_+^2 + p_-^2}{\rho_0 c} \quad (2.19)$$

donde

- p_+ es la onda de presión acústica que viaja en la dirección positiva de "x"
- p_- es la onda de presión acústica que viaja en la dirección negativa de "x"

Si la onda viaja en el sentido positivo de "x", la intensidad sonora instantánea es:

$$I = \frac{p_+^2}{\rho_0 c} \quad (2.20)$$

Y la intensidad sonora promedio es:

$$I = \frac{p_+^2}{2\rho_0 c} \quad (2.21)$$

Aplicando (2.17), a un campo de onda esférica, dado por las ecuaciones (2.15) y (2.16), encontramos que la intensidad sonora es⁶:

$$I = \frac{p_+^2 + p_-^2}{\rho_0 c} \cos\vartheta \quad (2.22)$$

donde

- p_+ es la onda de presión acústica que viaja en la dirección positiva de "x"
- p_- es la onda de presión acústica que viaja en la dirección negativa de "x"
- ϑ es el ángulo entre el vector velocidad de partícula y la velocidad de partícula en donde se hace la medición.

El ángulo ϑ es análogo al factor de potencia en circuitos eléctricos, solo que aquí quienes están en fase o no; son la presión y la velocidad de partícula. Cuando se incrementa la distancia radial r la onda puede aproximarse a una onda plana ya que, tanto la presión como la velocidad de partícula estarán en fase ($\vartheta=0$) por lo que si la onda viaja en el sentido positivo de "x", la intensidad sonora promedio será exactamente igual a la intensidad sonora promedio de la onda plana, ecuación (2.21).

Aunque la expresión (2.22) solo depende de una sola variable espacial, es claramente más complicada que para una onda plana, por que la relación de fases entre la presión y la velocidad de partícula, depende de la distancia.

Es obvio que las ondas de presión y de velocidad de partícula son de naturaleza armónica por lo que la intensidad sonora instantánea se puede dividir en dos componentes⁶: (i) *componente activa (o real)*, cuya promediación en el tiempo es diferente de cero, e indica la transportación neta local de energía sonora en dirección normal al frente de onda (se da en los campos sonoros activos); y (ii) *componente reactiva (o imaginaria)*, indica el transporte local de energía oscilatoria (se da en los campos sonoros reactivos), cuyo valor promedio es cero.

La componente activa se da en los campos de onda viajera (plana o esférica); la componente reactiva en los campos de onda estacionaria (plana o esférica); en la naturaleza los campos sonoros son complejos, e involucran parte activa y parte reactiva, una forma de visualizar esto es considerando un campo de presión unidimensional de una sola frecuencia, cuya ecuación es³:

$$p(x, t) = P(x) e^{i(\omega t + \phi_p(x))} \quad (2.23)$$

donde

$P(x)$ es la parte real dependiente del espacio, de la amplitud de presión.
 ϕ_p es la fase dependiente del espacio.

El gradiente de presión será entonces:

$$\frac{\partial p}{\partial x} = \left[\frac{dP}{dx} + i \left(\frac{d\phi_p}{dx} \right) P \right] e^{i(\omega t + \phi_p)} \quad (2.24)$$

De esta última ecuación obtenemos la velocidad de partícula, correspondiente a la ecuación de presión descrita antes, utilizando la ecuación de Euler.

$$u = \left(\frac{1}{\omega \rho_0} \right) \left[-P \left(\frac{d\phi_p}{dx} \right) + i \left(\frac{dP}{dx} \right) \right] e^{i(\omega t + \phi_p)} \quad (2.25)$$

La componente de la velocidad de partícula en fase con la presión es asociada con la componente activa de la intensidad, que para este caso se puede expresar como:

$$I_a(x, t) = - \left(\frac{1}{\omega \rho_0} \right) \left[P^2 \left(\frac{d\phi_p}{dx} \right) \right] \cos^2(\omega t + \phi_p) \quad (2.26)$$

Y su valor medio será:

$$I_a(x) = - \left(\frac{1}{2\omega \rho_0} \right) \left[P^2 \left(\frac{d\phi_p}{dx} \right) \right] \quad (2.27)$$

Resultado que nos indica la definición de intensidad sonora. La componente reactiva de la intensidad; esta dada por la siguiente ecuación:

$$I_{\text{reac}}(x, t) = - \left(\frac{1}{4\omega \rho_0} \right) \left[\frac{dP^2}{dx} \right] \sin 2(\omega t + \phi_p) \quad (2.28)$$

El valor medio de la componente reactiva de la intensidad, dada por la ecuación es cero. Como se ve en las dos últimas ecuaciones, la componente activa de la intensidad

es proporcional al gradiente espacial de la fase, y la componente reactiva es proporcional al gradiente espacial de la presión media cuadrada.

La intensidad sonora instantánea, en un campo sonoro unidimensional, se da por la suma de las ecuaciones (2.26) y (2.28), y puede ser escrita en la forma:

$$I(x,t) = I_a(x) \left[1 + \cos 2(\omega t + \phi_p) \right] + I_{reac}(x) \sin 2(\omega t + \phi_p) \quad (2.29)$$

donde

$$I_a(x) = \left[\frac{1}{2\omega\rho_0} \right] \left[P^2 \left(\frac{d\phi_p}{dx} \right) \right] \quad \text{y} \quad I_{reac}(x,t) = \left[\frac{1}{4\omega\rho_0} \right] \left[\frac{dP^2}{dx} \right]$$

además
$$p(x,t) = P(x) e^{i\phi_p(x)} e^{i\omega t}$$

Una expresión matemática más compacta para la ecuación (2.29), que es análoga, a la representación exponencial compleja de una cantidad armónica es:

$$I(x,t) = \text{Re} \left\{ C(x) \left[1 + e^{-2i(\omega t + \phi_p)} \right] \right\} \quad (2.30)$$

donde

$$C(x) = I_a(x) + iI_{reac}(x) = I(x) + iQ(x) \quad (2.31)$$

La ecuación (2.31) es conocida como 'intensidad compleja'. La parte real de C es la intensidad media (activa), y la parte imaginaria Q es la amplitud de la intensidad reactiva.

De aquí en adelante, se simbolizara al termino de intensidad media activa

$$I_a$$

con el símbolo I , que representa a la cantidad que es más extensamente medida, y corresponde a la definición más común de intensidad sonora, que indica la promediación en el tiempo de la razón de flujo de energía que pasa por una área unitaria del fluido.

La naturaleza compleja de C , es por el hecho de que los dos agentes de flujo de energía, presión y velocidad de partícula, no están necesariamente en fase. Considerando esto, y la forma explícita de la intensidad sonora, como el producto de la presión

$$p(x,t) = P e^{i\omega t} e^{i\phi_p} \quad \text{y de la velocidad de partícula} \quad u(x,t) = U e^{i\omega t} e^{i\phi_u}$$

donde ϕ_u es análogo a ϕ_p . Por lo que la intensidad se puede escribir como:

$$I(x, t) = \frac{1}{2} PU [\cos(2\omega t + 2\phi_p + (\phi_u - \phi_p)) + \cos(\phi_p - \phi_u)]$$

$$\begin{aligned} I(x, t) &= \frac{1}{2} PU [\cos(2(\omega t + \phi_p)) \cos \phi_r + \sin(2(\omega t + \phi_p)) \sin \phi_r + \cos \phi_r] \\ &= \operatorname{Re} \left\{ \frac{1}{2} PU e^{i\phi_r} \left[1 + e^{-2i(\omega t + \phi_p)} \right] \right\} \end{aligned} \quad (2.32)$$

donde $\phi_u - \phi_p$ esta reemplazado por ϕ_r . Por analogía con la ecuación (2.29)

$$C = I + iQ = \frac{1}{2} PU e^{i\phi_r} \quad (2.33)$$

$$|C| = \frac{1}{2} PU \quad (2.34)$$

$$I = \frac{1}{2} PU \cos \phi_r \quad (2.35)$$

$$Q = \frac{1}{2} PU \sin \phi_r \quad (2.36)$$

En un campo sonoro tridimensional la intensidad acústica tendrá la misma forma que para un campo sonoro unidimensional, solo que en lugar de tener una sola coordenada (x) tendrá un vector de posición " r ", entonces la ecuación (2.31) se puede expresar en la siguiente forma

$$C = \frac{1}{2} P \left[U e^{i(\phi_p - \phi_r)} \right]_i + V e^{i(\phi_p - \phi_v)} \left[\right]_j + W e^{i(\phi_p - \phi_w)} \left[\right]_k \quad (2.37)$$

Una forma análoga de obtener la intensidad compleja, y que es muy común en la practica, es como sigue⁷:

$$C = \frac{1}{2} PU^* \quad (2.38)$$

$$I = \frac{1}{2} \operatorname{Re}\{PU^*\} \quad (2.39)$$

$$Q = \frac{1}{2} \operatorname{Im}\{PU^*\}$$

donde

U^* es el complejo conjugado de la velocidad de partícula, y se emplea para obtener la magnitud de la intensidad activa o reactiva.

En este proyecto de tesis la cantidad que se mide, es la intensidad promedio en el tiempo, es decir la intensidad activa (en una sola dirección del vector velocidad de partícula), para esto es necesario contar con el instrumental adecuado para dicha medición de esta cantidad física. En el siguiente subcapítulo, se mencionaran los métodos de medición de esta cantidad. Los instrumentos empleados son: la sonda de intensidad sonora (conocida como probador de intensidad sonora), y un analizador de frecuencias, siendo este último instrumento el que se encarga de integrar la información captada por la sonda con la finalidad de obtener la intensidad sonora.

2.1 METODOS DE MEDICION DE INTENSIDAD SONORA

Todo sistema de análisis de intensidad sonora consiste de una sonda (o probador) y de un analizador de frecuencias, básicamente. La sonda simplemente mide la presión acústica y la velocidad de partícula. Por lo que la tarea del analizador es el procesamiento de las variables medidas y de los cálculos necesarios para encontrar la intensidad sonora.

De acuerdo con la definición de intensidad sonora instantánea, como el producto de la presión acústica instantánea y la velocidad de partícula instantánea, cualquier sistema de medición de intensidad, en principio, tiene que incluir transductores para cada una de las dos variables, antes mencionadas, los cuales estarán integrados de tal manera que conformen, una sonda (o probador).

Existen dos categorías de sondas ó probadores de intensidad, que son ampliamente usados; una combina, un transductor de presión con una unidad de transducción de velocidad de partícula; la otra comprende, dos transductores de presión (micrófonos), nominalmente idénticos. A la primera categoría se le conoce como probadores del tipo 'p-u', y a la segunda categoría como probadores del tipo 'p-p'.

Principio de transducción 'p-u'

Las señales que entrega esta sonda ó probador, que incorpora una combinación de un transductor de presión con un transductor de velocidad de partícula, son multiplicadas para obtener la componente dependiente del tiempo, de la intensidad, en la dirección del eje de la sonda. Actualmente los probadores disponibles de este tipo, combinan micrófonos de condensador con transductores ultrasónicos de velocidad de partícula.

Como se puede ver en la figura 2.1.1, dos rayos ultrasónicos paralelos, son lanzadas en direcciones opuestas; la transmisión de estas ondas por la oscilación del aire en movimiento, cuando una frecuencia de audio esta presente, producen una diferencia de fase entre las dos ondas ultrasónicas que llegan a sus respectivos receptores. Esta

diferencia de fase es análoga a la componente de la velocidad de partícula de la onda de audio frecuencia, en la dirección de los rayos de ultrasonido.



Fig 2.1.1. Sonda de intensidad 'p-u' tipo 216 de Norwegain Electronics.

Por supuesto este sistema también responde a los movimientos en el aire de tipo no acústico, tal como la turbulencia del viento, por lo que se deben tomar precauciones, en las mediciones.

También en la figura 2.1.1, se puede ver en el centro del aro, al micrófono de condensador que nos proporcionara la presión acústica.

En ausencia de flujo de aire sobrepuesto, el transito a lo largo del tiempo de cada rayo es el mismo, y esta dado por:

$$t_0 = \frac{d}{c} \quad (2.1.1)$$

donde

d es la distancia entre las caras del emisor y el receptor.

c es la velocidad de propagación de sonido en el medio.

Si un flujo estable de velocidad u es sobrepuesto, la transición en el tiempo llega a ser:

$$t^+ = \frac{d}{(c+u)} \quad (2.1.2) \quad \text{y} \quad t^- = \frac{d}{(c-u)} \quad (2.1.3)$$

Por lo que el resultado de la diferencia de fases es³:

$$d\phi = \omega_u d \left[\frac{1}{(c-u)} - \frac{1}{(c+u)} \right] \approx \frac{2\omega_u du}{c^2} \quad \text{si } u \ll c \quad (2.1.4)$$

donde

ω_u es la frecuencia ultrasónica

Esta diferencia de fases es convertida a una cantidad eléctrica, análoga a u . La señal proporcional a la presión puede ser multiplicada directamente con la señal proporcional a la componente de la velocidad de partícula, para producir una cantidad análoga a la intensidad sonora instantánea. Esta cantidad puede ser promediada en el tiempo para obtener la componente de intensidad media (activa). La descomposición en la frecuencia de la intensidad se puede alcanzar por el filtraje de las dos señales antes de ser multiplicadas, o por el empleo de un análisis espectral. El vector de intensidad sonora instantánea total $I(t)$ puede ser determinado solo si las tres componentes ortogonales de la velocidad de partícula son simultáneamente medidas.

Principio de transducción 'p-p'

Dos transductores de presión (micrófonos) nominalmente idénticos son colocados juntos en un soporte que está diseñado para minimizar la difracción de los campos sonoros incidentes. Los transductores normalmente son micrófonos de condensador de alta calidad, para mediciones en el aire, e hidrófonos piezo eléctricos para mediciones en el agua.

Estos micrófonos de condensador pueden ser asociados en varias configuraciones³, 'cara a cara', 'lado a lado', 'cascada', 'cola a cola', como se puede ver en la figura 2.1.2

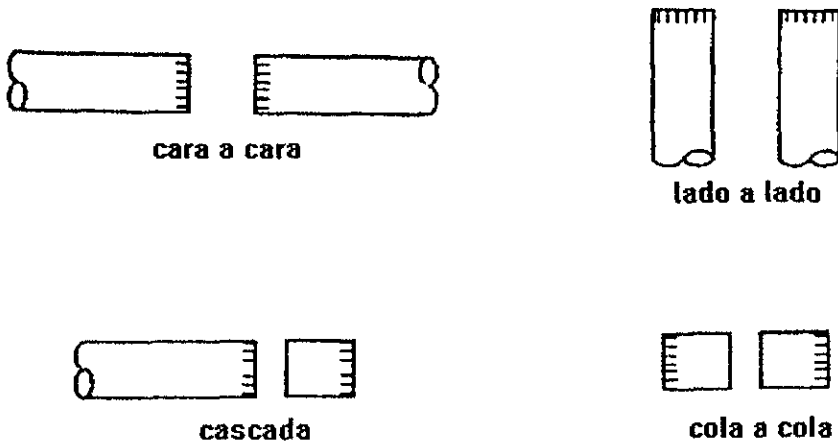


Figura 2.1.2 Esquema que muestra las diferentes configuraciones para las sondas del tipo 'p-p'.

La velocidad de partícula es obtenida con esta sonda por medio de la ecuación de Euler, ecuación (2.7) que nos dice que.

$$\nabla p(t) + \rho_0 \frac{\partial u(t)}{\partial t} = 0 \quad (2.1.5)$$

La ecuación (2.1.5) puede ser expresada como el gradiente de presión en una dirección " r " proporcional a la componente de la aceleración de la partícula en esa dirección:

$$\frac{\partial p}{\partial r} = -\rho_0 \frac{\partial u_r}{\partial t} \quad (2.1.6)$$

Por lo que de (2.1.6) se obtiene la velocidad de partícula, dada por la siguiente integral:

$$u_r(t) = -\left(\frac{1}{\rho_0}\right) \int_{-\infty}^t \left(\frac{\partial p(t)}{\partial r}\right) dt \quad (2.1.7)$$

La ecuación (2.1.7) puede ser expresada de otra forma por medio de una aproximación de diferencia finita, ya que en la práctica el gradiente de presión puede ser aproximado por la diferencia de presiones captadas por dos micrófonos p_A y p_B , estando espaciados uno de otro (una distancia Δr), y dividiendo esta diferencia entre la distancia de separación. Esta aproximación es válida si la longitud de la separación es pequeña comparada con la longitud de onda, del sonido medido, ($\Delta r \ll \lambda$).

$$u_r(t) \approx -\left(\frac{1}{\rho_0 \Delta r}\right) \int_{-\infty}^t [p_B(t) - p_A(t)] dt \quad (2.1.8)$$

donde

p_A y p_B	son las señales de presión en los dos micrófonos
Δr	es la distancia que separa los centros acústicos de los transductores
$u_r(t)$	es el vector velocidad de partícula en la dirección r

Ahora la intensidad sonora se puede obtener multiplicando la ecuación (2.1.8) de la velocidad de partícula por la presión acústica. La presión acústica se estima de la presión media entre los dos micrófonos:

$$p(t) \approx \frac{p_A(t) + p_B(t)}{2} \quad (2.1.9)$$

Entonces la intensidad sonora en la dirección r , que corresponde a la intensidad activa, se expresa con la siguiente ecuación⁵:

$$I_r = -\frac{1}{2\rho_0 \Delta r} \langle (p_A + p_B) \int (p_B - p_A) dt \rangle = I_a \quad (2.1.10)$$

donde

$\langle \rangle$ es el operador de promediación en el tiempo.

Conociendo los fundamentos de cómo se mide la intensidad sonora y antes de entrar en detalles sobre el método empleado en este trabajo, es necesario conocer primero que aplicaciones tiene, y en base a esto, se describirá el método que se utilizó.

2.2 APLICACIONES

La técnica de medición de intensidad sonora tiene básicamente dos ventajas. Primero, responde únicamente, al flujo neto de energía sonora; rechazando los efectos debidos a la parte reactiva de los campos sonoros, como las que poseen los campos difusos, o por la presencia de ondas estacionarias.

La segunda ventaja que tiene esta técnica, es que en esta medición se obtiene información direccional, por lo que no únicamente nos indica el flujo neto de energía sonora, si no que también nos indica de donde proviene este flujo. Estas dos ventajas son de suprema importancia cuando se trata de identificar fuentes sonoras. Igualmente importante, es el hecho de que las mediciones de intensidad, sean la base para calcular la potencia sonora emitida por una fuente, independientemente del ruido de fondo. En pocas palabras, cuando las mediciones de intensidad sonora son usadas, se puede medir potencia sonora en un medio ambiente ruidoso, sin tener que trasladar a dichas fuentes sonoras a otros lugares que nos dan facilidades para su estudio, como lo son las cámaras anecoica y reverberante.

Cabe recordar que:

$$\text{Intensidad} = \frac{\text{Potencia}}{\text{Area}} \left[\frac{W}{m^2} \right]$$

De la igualdad anterior, es posible determinar la potencia acústica, al multiplicar la intensidad por el área de una superficie S , así que la siguiente ecuación nos da la potencia neta que entrega una fuente sonora.

$$\int_S I(t) \cdot d s = \Pi [W] \quad (2.2.1)$$

Donde s es el área de una superficie que encierra a la fuente³, Ver figura 2.2.1. Ya sea para la detección de fuentes sonoras ó bien para el cálculo de la potencia acústica que entrega cualquier fuente, es necesario cubrir a esta, con una superficie, que puede ser plana o hemisférica, en el caso de la medición de potencia acústica necesariamente la fuente tiene que estar dentro de una superficie hemisférica ó bien cubica. Para el caso de detección de fuentes, únicamente se requiere de cubrir a la fuente con un plano paralelo a dicha fuente, ver figura 1.1

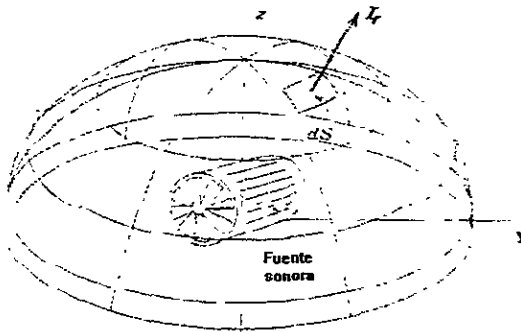


Figura 2.2.1 Indica como se mide la intensidad sonora en una superficie hemisférica, para el cálculo de la potencia acústica de la fuente contenida dentro de dicha superficie.

Al cubrir con una superficie la fuente en estudio, se tiene que calcular la intensidad que pasa toda esa área. Existen dos formas de hacer esto⁶: cubriendo la superficie con la sonda en un barrido en forma discreta, y la segunda es hacer un barrido en forma continua. Ver figura 2.2.2

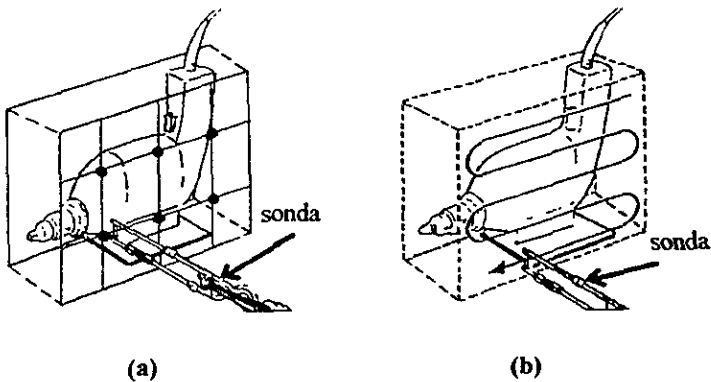


Figura 2.2.2 Se muestran los tipos de barridos que se pueden realizar para medir intensidad sonora, en (a) se tiene un barrido en forma discreta, al poner a esta en los puntos de unión de una rejilla, que puede estar construida por hilos ó alambres, (b) es un barrido en forma continua, que se realiza con un movimiento suave en zig zag.

Para el barrido en forma continua, es necesaria una promediación en el tiempo adecuadamente grande, la sonda es simplemente movida sobre la superficie en zig zag,

como si se estuviera pintando a dicha superficie. A de más, el barrido sobre la superficie tiene que ser a una razón constante, y se tiene que cubrir la superficie de una manera uniforme. Este método entrega un sólo valor de intensidad promedio.

El método, de barrido en forma discreta, divide el área en pequeños segmentos, y mide la intensidad en cada segmento. Los puntos de medición son marcados con una rejilla.

Las formas de realizar el barrido son dos: manualmente o con un robot. La primera forma, se puede decir que es la más económica y fácil, por que no se requiere de equipo extra para efectuar este barrido, pero requiere de mucho tiempo para ir midiendo, una forma de optimizar el tiempo es el empleo de cuando menos dos personas (lo que encarece esta forma de medición), una se encarga de colocar la sonda y otra de controlar el equipo de medición. Aun que existen equipos que son controlados por control remoto por una sola persona se sigue desperdiciando tiempo en cubrir la zona de medición.

El empleo de un robot optimiza tiempo, mientras el robot coloca automáticamente a la sonda el operador puede ir registrando las mediciones (o realizar cualquier otra actividad si se cuenta con un sistema totalmente automatizado que coloque la sonda, registre las mediciones y muestro los resultados), con esta forma de barrido se gana precisión, y capacidad para poder repetidamente hacer barridos con la misma precisión. Su gran desventaja es que dicho robot puede ser muy caro.

Ningún método es mejor en todas las aplicaciones, en algunos casos ambos métodos son útiles. La técnica de barrido continuo, es matemáticamente una mejor aproximación a una integral continua en el espacio y es a menudo más exacta. Pero necesita mayores cuidados el movimiento de la sonda, que tiene que tener una cadencia constante para cubrir la superficie igualmente. El método de barrido en forma discreta, es más repetible, con la misma exactitud y precisión en la colocación de la sonda en los puntos de medición.

La automatización de ambos métodos, mejora la exactitud y la precisión en las mediciones, con lo que obtienen resultados que por mucho son más exactos, además de que se consigue barrer cualquier superficie en forma repetida, al eliminar los errores inherentes a la persona que se encarga de hacer el barrido.

2.3 METODO DE MEDICION EMPLEADO

Como se mencionó anteriormente, este trabajo de tesis tiene como finalidad obtener los mapas de intensidad sonora de una guitarra clásica, conociendo los fundamentos y los métodos de medición y delimitados por el equipo disponible en la sección de acústica del Centro de Instrumentos, se optó por, diseñar un sistema mecánico (robot) que efectuase el barrido en forma discreta (los detalles de este 'robot', se discuten en el capítulo 3). Hablamos de limitaciones impuestas por los equipos disponibles entre los que se encuentran; el tipo de analizador empleado y la sonda que se utilizó, el analizador utiliza un método por filtros digitales para descomponer la señal acústica en varias bandas de frecuencia, la sonda disponible utiliza el método de transducción 'p-p'.

Se trata de una sonda de intensidad sonora tipo 'p-p', con una configuración cara a cara y espaciados por un separador sólido. Como ya se mencionó en la sección anterior este tipo de sonda utiliza un método llamado de aproximación de diferencias finitas, el cual aproxima el gradiente de presión a diferencias de presión que captan los dos micrófonos de la sonda, y dividir este valor entre la distancia de los micrófonos.

En la figura 2.3.1 se puede ver el método de aproximación de diferencias finitas que se utiliza para aproximar el gradiente de presión, para una configuración cara a cara y para una configuración lado a lado. De dicha figura se puede obtener la ecuación (2.1.11), con dirección r .

Antes de continuar hay que mencionar las unidades en que se trabaja en este proyecto, las cuales son:

$$I = 20 \log \left(\frac{I}{I_{ref}} \right) \quad [dB \text{ ref } I_{ref}]$$

Entonces las unidades para el nivel de intensidad son en dB, referidos a 10^{-12} $[W/m^2]$ de intensidad.

Ahondando más en este tipo de sondas, es obvia la necesidad de conocer la mejor configuración y el espaciamiento óptimo entre micrófonos. La configuración de los micrófonos es importante, por que cada micrófono tiene influencia en el campo sonoro que ve el otro micrófono. En la práctica la mejor configuración es la 'cara a cara' por que el arreglo es más simétrico, además con un separador sólido, que evita la difracción, de las ondas sonoras⁵ (esta configuración es usada en la sonda empleada en este proyecto y que es la sonda de intensidad sonora tipo 3545 de la marca Brüel & Kjær).

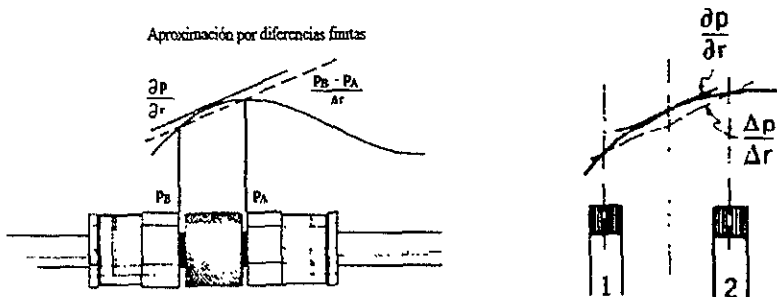


Figura 2.3.1 Aproximación por diferencias finitas, para una configuración cara a cara (izquierda) espaciados por un elemento sólido, y para una configuración lado a lado (derecha), sin ningún tipo de espaciador.

Ahora, bien la utilización de un separador sólido en la configuración cara a cara, proporciona el mejor control en la separación de los micrófonos, esto es importante ya

que el equipo que se encarga de integrar y procesar la información obtenida por la sonda, tiene que asumir, que la distancia Δr es equivalente a la distancia física que tiene el separador.

El espaciador ó separador sólido es muy importante, ya que del depende si una medida será realmente aceptable. Una limitación que puede causar un error en la medición, por el método de aproximación de diferencias finitas, es que existen límites de respuesta en la frecuencia para que una lectura sea aceptable, así que el límite para alta frecuencia, o error de alta frecuencia es también llamado error de *bias* (error de sesgo) que da una falsa estimación de la intensidad medida. La sonda como ya sabemos aproxima el gradiente de presión de una curva, a una línea recta entre dos puntos. Si los cambios de la curva son demasiados rápidos con respecto a la distancia, la estimación será por lo tanto inadecuada. Esto pasará si la longitud de onda medida, es pequeña comparada con la separación efectiva de los micrófonos, ver la figura 2.3.2.

Para cualquier separación efectiva entre los micrófonos siempre habrá un límite en alta frecuencia que hará que los errores aumenten significativamente. Para que una lectura tenga una exactitud entre 1dB, la longitud de onda medida tiene que ser seis veces más grande que la longitud del separador. Que corresponde a los siguientes límites en alta frecuencia⁴:

50mm:	hasta 1.25 KHz
12mm:	hasta 5 KHz
6mm:	hasta 10 KHz

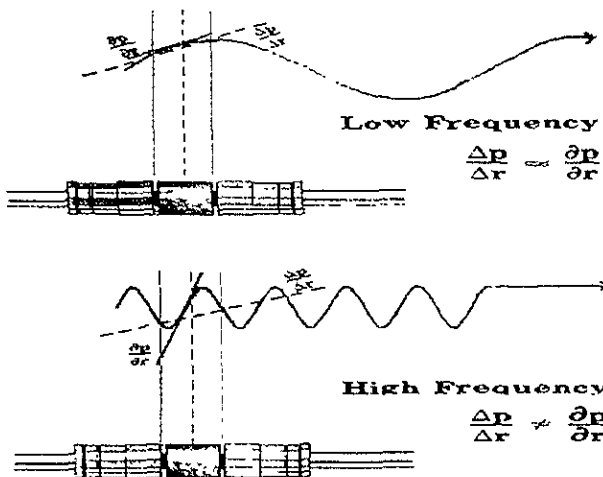


Figura 2.3.2 Limitación en el método de aproximación de diferencias finitas.

Una vez mencionados algunos de los detalles de la sonda de intensidad sonora utilizada; pasamos a comentar que el método de barrido que se utiliza, es el de barrido en forma discreta. El porque se eligió este tipo de barrido y no el barrido en forma continua, obedeció a los requerimientos de los investigadores que trabajan en este centro, uno de estos requerimientos, indicaba, que se tenían que elaborar los mapas a partir de un barrido en forma discreta. Como se mencionó en las secciones anteriores, la intensidad sonora nos indica el flujo neto de energía sonora que fluye a través una área determinada, esa área puede valer $L \cdot M$, según la figura 2.3.3, en dicha figura se puede ver una rejilla que representa un barrido en forma discreta, los puntos de unión representan los lugares en que se tiene que colocar la sonda para las mediciones, una vez obtenidas las intensidades correspondientes a cada punto de unión, se interpolan los datos a fin de obtener el mapa de intensidad sonora correspondiente al área $L \cdot M$, en sus diferentes formatos de presentación.

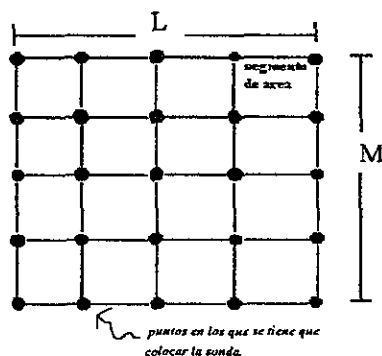


Figura 2.3.3 Representación de una rejilla con 16 segmentos de área y 25 puntos de medición.

El área a la que se desea obtener el mapa de intensidad puede ser tan grande como se quiera, de esta forma es posible obtener los mapas de intensidad para un avión o para un trailer, obviamente se pierde resolución, a un que para este tipo de fuentes sonoras que rebasan más de los 120dB no es tan significativo.

Por sentido común se elige el tamaño de los segmentos de área (separación entre puntos de la rejilla), así pues, estos tienen que ir de acuerdo con el tamaño de la fuente.

El cómo, se hace la colocación de la sonda en los puntos de unión de una rejilla (imaginaria) en forma automática, el cómo se adquieren los datos y la manera en que se integran estos, y el procesamiento de estos para encontrar los mapas de intensidad sonora, se mencionan en los siguientes capítulos.

Para concluir con este tema, es necesario mencionar que el analizador de frecuencias empleado en este proyecto fue el 2133 de la marca Brüel & Kjær, que utiliza una técnica de descomposición espectral, en la banda audible (20 Hz a 20KHz), con la técnica de filtros digitales, los detalles de la sonda y del analizador empleados, se comentan en el capítulo 4.

REFERENCIAS CAPITULO 2

1. E. Skudrzyk, *The foundations of acoustic*, (Ed. Springer - verlag, New York),(1971)
2. L. E. Kinsler, A. R. Frey, A. B. Coppens, *Fundamentos de acústica* (Ed. Limusa, México), (1988)
3. F.J. Fahy, *Sound Intensity* (Ed. Elsevier, New York), (1989)
4. Brüel & Kjær, *Sound Intensity*, Technical Documentation.
5. R. Upton, *Sound intensity-A powerful new measurement tool*, Sound and Vibration Oct, 10-18, (1982)
6. M. J. Crocker, F. Jacobsen, *Sound intensity*, Encyclopedia of acoustics (Ed. John Wiley & Sons), (1997)
7. F. Cervera, F. Belmar, H. Etelles, *A low cost robot for acoustic intensity measurements*, Noise Control Eng. J. **42(1)**, 31-35, (1994)

CAPITULO 3

3. SISTEMA DE POSICIONAMIENTO Y ORIENTACION DE LA SONDA

En este capítulo se abordaran los detalles de construcción del sistema de posicionamiento y orientación, que se controlan mediante la computadora, quien se encarga de posicionar la sonda en los puntos de interés a medir. A este sistema lo he llamado robot. Se trata de un robot con dos grados de libertad (x,y), que consigue el movimiento de su carro porta sonda, con dos motores de corriente directa (uno por cada eje), los cuales se controlan a través del puerto paralelo de la computadora y de circuitos de interfaz entre la PC y el robot. Este sistema de posicionamiento y orientación esta compuesto por lo siguiente

- Sistema mecánico.
- Circuitos de control (control de motores con el puerto paralelo)
- Circuitos para conocer la posición (tarjetas Pc-Lab e interfaz para conocer posiciones con dos hilos).
- Programas de control (hechos en lenguaje C ver 7)

Este robot es capaz de cubrir con su carro (porta sonda), en forma discreta, una área de 0.3072 m^2 que se deriva de la figura 3.1, que es una rejilla imaginaria.

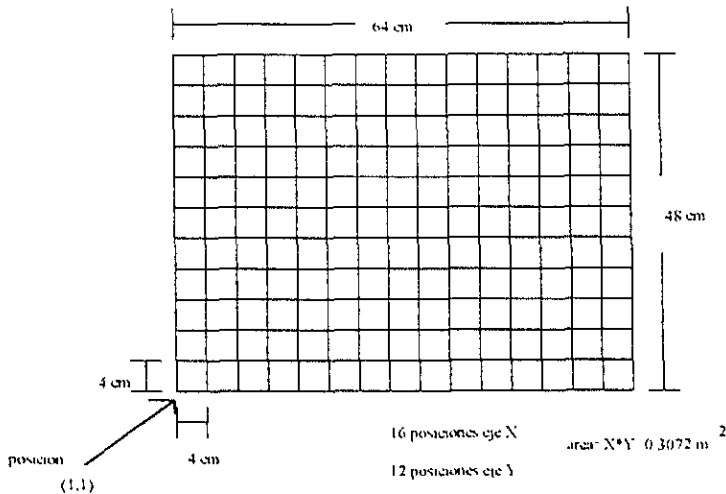


Figura 3.1 Area que cubre el robot

El robot puede, colocar a la sonda en 192 puntos de medición correspondientes a 165 segmentos de área, derivados de las 16 posiciones en X y de las 12 posiciones en Y . Este a su vez se puede mover libremente cubriendo cualquier segmento de área, en forma de zigzag en dirección vertical o dirección horizontal, comenzando siempre de la posición (1,1), por ejemplo se le puede ordenar (con los programas de computadora) cubrir en forma de zigzag horizontal el área que va del punto (1,1) hasta el punto (8,5), ver figura 3.2. Para lograr lo anterior, la computadora tiene que conocer la posición actual del robot, conocer la posición a la que mandará al robot y finalmente activar los motores que provocan el movimiento.

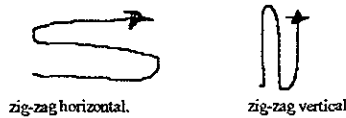


Figura 3.2 Dirección del barrido.

Adelantándose un poco, ya que el programa WT9378 (programa para generar mapas de intensidad sonora, con las señales medidas por el anlizador 2133), únicamente permite un barrido con 16 posiciones en X y 12 en Y . Aun siendo que el robot diseñado permite el movimiento sobre una rejilla de 16 posiciones en X y 16 posiciones en Y , es decir, el área que puede cubrir este robot es de 0.4096 m^2 , por las limitaciones del programa se cubre solo un área de 0.3072 m^2 . Queriendo aprovechar al máximo los circuitos para conocer posiciones es que se decidió hacer un robot de 16 por 16 posiciones, que puede tener otras aplicaciones ya que el robot, su programa y sus circuitos de control tienen un funcionamiento independiente del programa WT9378.

Finalmente y antes de entrar en detalles de construcción, se menciona brevemente, que el sistema mecánico, son todas las partes, principalmente metálicas, que conforman el robot. Los programas de control son los que se encargan de dar instrucciones a los circuitos de control para que los motores realicen una función específica, a fin de proporcionar movimiento al robot. A su vez estos programas reciben información captada por los sensores de posición y los circuitos de interfaz correspondientes, para que el robot al mover su carro porta sonda, tenga la capacidad de colocarla en el punto deseado. Ver figura 3 3

3.1 SISTEMA MECANICO

El sistema mecánico, es la parte que conforma el medio para conseguir el transporte de la sonda, mediante los movimientos de las piezas mecánicas necesarias, para efectuar dicha acción.

Este sistema mecánico, simplemente esta compuesto por dos rieles de aluminio montados en forma de cruz y que representan, tanto al eje X como al Y . Sobre el riel X , esta montado un carro, que se mueve a lo largo de dicho eje, sobre de este carro esta montado el riel del eje Y . También dentro de este riel Y esta montado otro carro (carro

porta sonda), que se puede mover a lo largo de dicho eje, sobre de este carro porta sonda esta la "cabeza" (la pieza para fijar esta ultima), ver figura 3.3. Las longitudes de los rieles que sirven, de eje X y de eje Y son: 80 cm para el eje X y de 80 cm para el eje Y. Por cada eje se tienen 16 cm de riel, que es más de lo necesario (se necesitan 64 cm para contener 16 posiciones con una separación de 4 cm entre posiciones por cada eje), los cuales se utilizan para la fijación de los rieles en los tripies y para dar a los carros una pequeña zona de seguridad para que estos puedan tomar las posiciones de cada extremo.

Los carros antes mencionados, consiguen moverse gracias a una tracción mediante poleas. ver figura 3.6.

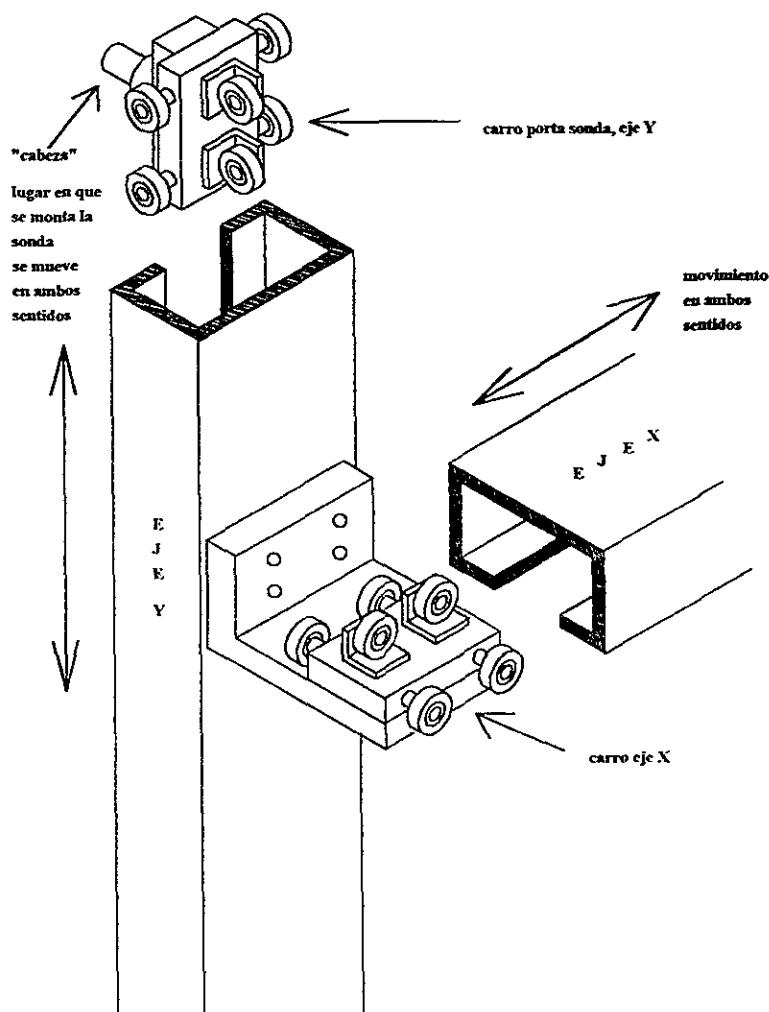


Figura 3.3 Se ilustra como esta compuesto el sistema mecánico

Para llegar a la conformación de la parte mecánica se ensayaron otras configuraciones, una de las cuales se muestra en la figura 3.4

La configuración de la figura 3.3 fue la que demostró más estabilidad mecánica y menor demanda de carga a los motores, debido a que se distribuye mejor la carga en el eje X (quien soporta el peso del eje Y , a más del peso de la sonda y de su unidad de control remoto) y reduce más las fuerzas de fricción. Por lo que se consideró al sistema de la figura 3.3, como óptimo para el proyecto.

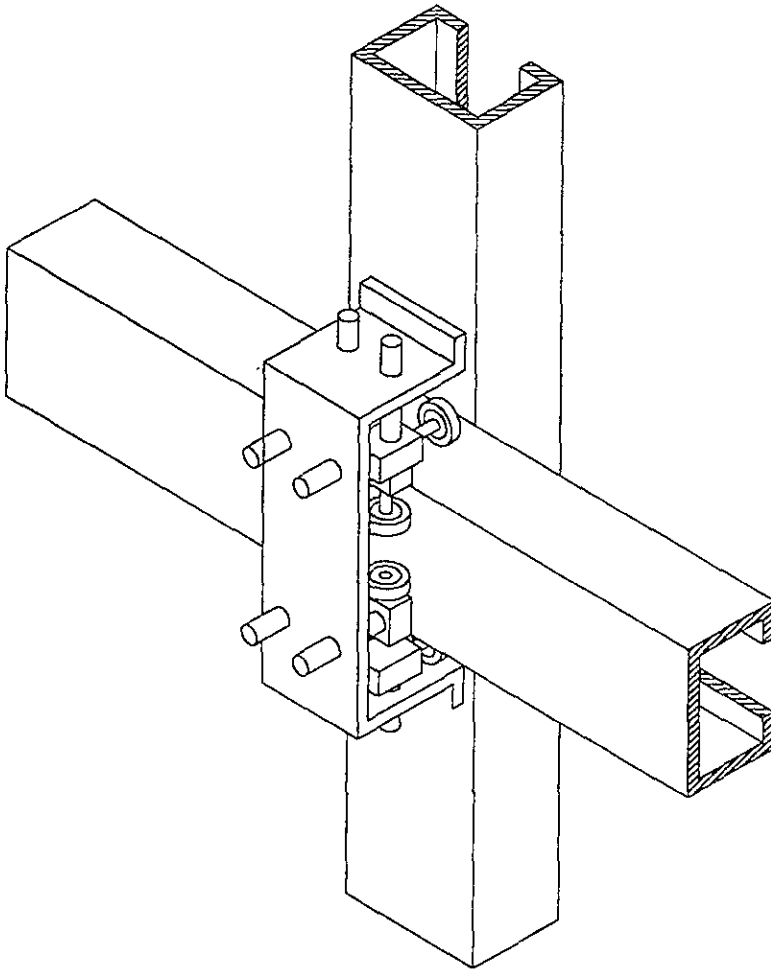


Figura 3.4 Esta configuración presento muchos problemas, por lo que fue descartada para la utilización final en el proyecto.

Los carros que se pueden ver en la figura 3.3, están hechos con placas y barras de aluminio, con baleros y con ruedas de goma. Las ruedas de goma se utilizan para ejercer una fuerza entre el carro y el riel que lo contiene, gracias a su capacidad de deformación hacen que los carros se mantengan en posiciones fijas.

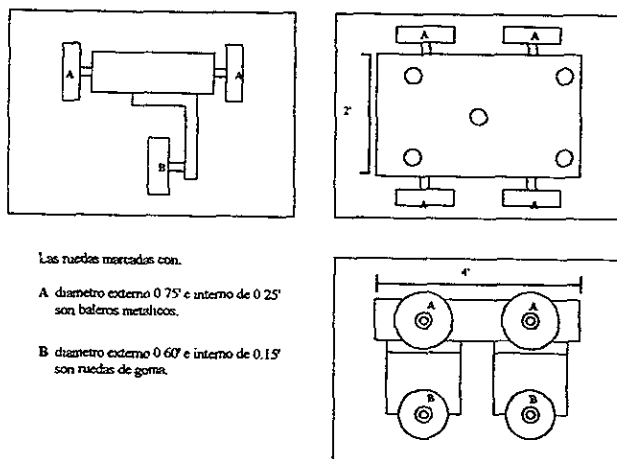


Figura 3.5 Se muestran las medidas para los carros del eje X y del eje Y.

Para el carro del eje Y únicamente se le agregaron otras piezas que sirven para atornillar a la sonda junto con su unidad de control remoto, dichas piezas forman el carro porta sonda.

Como se mencionó anteriormente la tracción es mediante poleas, una en la flecha del motor y otra fija, como se muestra en la figura 3.6

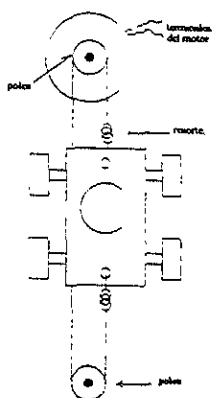


Figura 3.6 Se muestra la tracción utilizada

Para el montaje del sistema mecánico es necesario la utilización de soportes ó tripies, en la figura 3.7 se puede ver el montaje completo.

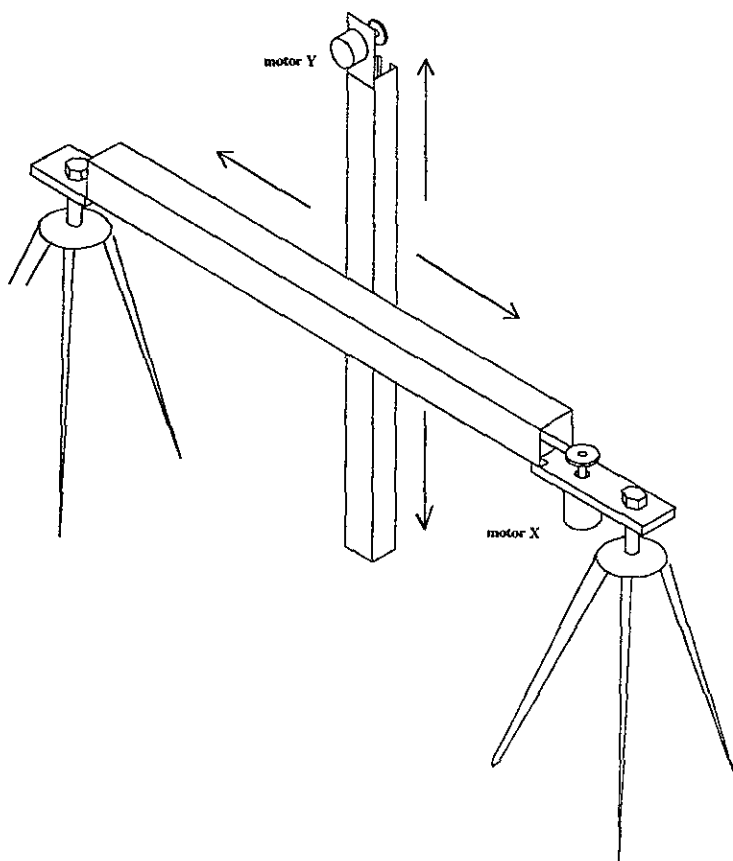


Figura 3.7 Montaje completo del robot.

Para eliminar ruidos, que pudiesen mezclarse con las ondas sonoras del objeto en estudio, y para reducir efectos de inestabilidad se colocaron tiras adhesivas de material para aislamiento acústico y de absorción de vibraciones, en los puntos más propensos a producir estos efectos

Al inspeccionar la figura 3.7, se puede observar, que los motores son los encargados de proporcionar la fuerza de tracción a los carros que mueven los ejes, así que enseguida se detallará como se cambia de una posición a otra, al sistema mecánico

3.2 POSICION

En el capítulo anterior se precisó que el barrido automático, es en forma discreta, lo que nos da ventajas, sobre un barrido continuo, por que no se requiere pasar de una posición a otra, con una velocidad uniforme. Por lo tanto, no se requiere de motores muy precisos y mucho menos rápidos, siendo su control muy sencillo de diseñar. Adicionalmente esta técnica de medición se encarga de capturar datos cuando la sonda se encuentra estática en el punto de interés y no cuando se esta en pleno movimiento (situación que no representa mayores dificultades para el diseño del sistema mecánico). Aun que el sistema mecánico, puede permitir la implementación de un barrido del tipo continuo, tomando en cuenta diversos detalles, como pudiesen ser, velocidades relativas uniformes, estabilidad a las oscilaciones de los elementos mecánicos, etc. la demanda de este proyecto, fue la de cubrir la superficie con un barrido discreto.

Para proveer al sistema mecánico con movimiento fue necesario estudiar las características de los motores de pasos y de los motores de corriente continua, además de la realización de programas de prueba, que junto con sus respectivos circuitos electrónicos, proporcionaran los medios de control, para que se cumplieran los propósitos y objetivos que demando este proyecto.

3.2.1 Motores de pasos

Se trata de motores ampliamente usados en mecanismos y robots, sus principales virtudes son la alta precisión, bajo consumo de corriente, supresión de retroalimentación en la determinación de la posición del eje y su bajo costo. El único problema que presentan, es su potencia limitada.

Este tipo de motores son capaces de transformar pulsos eléctricos (información digital) en movimientos mecánicos. El eje del motor gira un determinado ángulo por cada pulso de entrada. El resultado de este movimiento, fijo y repetible, es un posicionamiento preciso y fiable.

Un motor de pasos puede girar, en ambos sentidos, un número exacto de grados, con incrementos mínimos determinados por su diseño. Este incremento queda comprendido entre 0.72° y 90° , correspondientes a 500 y 4 pasos por revolución, respectivamente.

Existen dos modos de funcionamiento¹.

- Paso completo ("Full-step mode")
- Medio paso ("Half-step mode")

En el funcionamiento en paso completo, el motor realiza el paso en cuanto la secuencia adecuada se transfiere a los embobinados. Después de cuatro pasos, se conectan nuevamente los dos primeros embobinados, repitiéndose la secuencia.

La velocidad del motor es proporcional a la frecuencia con la que se envían los pulsos de excitación a los devanados.

En el funcionamiento a medio paso, el motor no hace una secuencia de cuatro pasos como el otro modo, ahora hace ocho, bajo estas condiciones el rotor gira la mitad de su recorrido normal por paso. Las ventajas de trabajar con medio paso consisten en una resolución mayor, con reducción de la resonancia y una velocidad mayor.

En los motores de pasos, a medida que aumenta la velocidad disminuye la corriente y, por lo tanto, el par motor también disminuye. Esta característica se tiene que tomar muy en cuenta para la selección de un motor.

Como uno de los propósitos de este proyecto fue el de realizar un sistema de bajo costo y con componentes disponibles en el mercado nacional. Puesto que los motores de pasos reúnen éstas características, se pensó utilizarlos en este proyecto, por lo que se realizaron programas de control y circuitos electrónicos para el control de estos motores.

Entrando en detalles, se tiene que por medio de la programación y los dispositivos electrónicos adecuados es posible controlar el movimiento de un motor de pasos de cuatro fases y un estator. Por lo que se diseñó e implementó un circuito de prueba.

El motor de pasos que se empleó es el 42SH-32KCA (Edmund Scientific), motor de cuatro fases con un estator, voltaje de alimentación: 9 volts de dc, corriente por fase: 0.125 amperes, impedancia: 60 ohms, paso angular: 3.75 deg.

Para generar la secuencia necesaria para producir movimiento se empleó el circuito integrado (CI) SAA1027, el cual fue conectado a la computadora, en su puerto paralelo. El programa fue compilado en C.

El programa entrega, por el bit menos significativo del puerto paralelo en su bus de datos, un flanco de reloj (generado por programación) a una etapa de aislamiento óptico entre la PC y los dispositivos que excitan al motor de pasos, y esta etapa a su vez entrega este flanco a la etapa de disparo del SAA1027. En el mismo bus de datos del puerto paralelo, por el quinto bit, se entrega un nivel lógico bajo a la etapa de aislamiento óptico, compuesta por optoacopladores 4N25 con salida a transistor NPN y conectados en forma inversora para invertir el nivel lógico y lograr el sentido de giro del SAA1027. Con un nivel lógico alto (niveles TTL) se logra que el motor gire en sentido opuesto a las manecillas del reloj.

El circuito integrado SAA1027 proporciona como máximo 0.350 A en modo de paso completo.

La alimentación de la tarjeta de control del motor, de la tarjeta de potencia y del mismo motor es de 9 volts de DC. Y el circuito se puede ver en la figura 3.2.1.

Una vez hecho el circuito de prueba y su montaje en la parte mecánica, se observó que este tipo de motores no proporcionan la fuerza suficiente para poner en movimiento al robot. Por lo que se llevó a cabo la alternativa de diseñar diversas configuraciones de trenes de engranes para conseguir mayor torque. Se consiguió lo anterior pero se sacrificó mucha velocidad, si el motor empleado es lento por sí mismo, con el tren de engranes esto se convirtió en crítico.

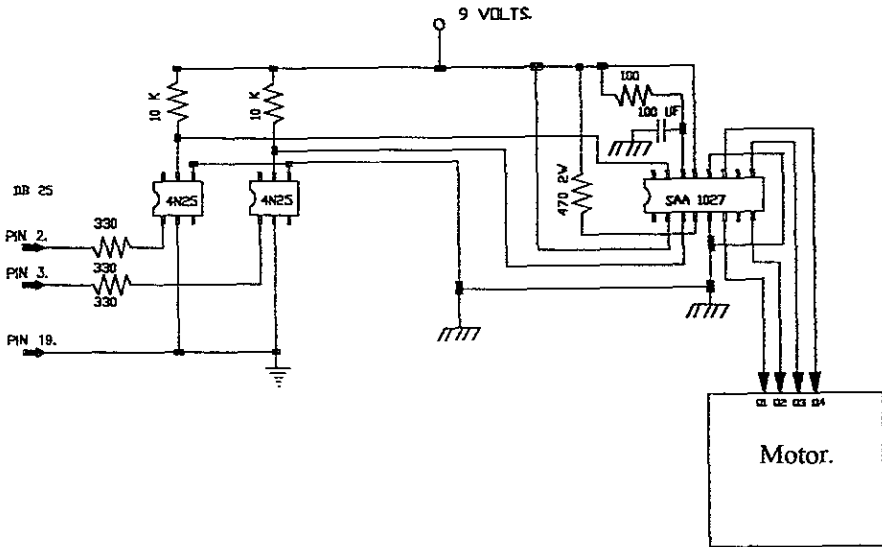


Figura 3.2.1 Circuito de prueba para el control de motores de pasos, pata2 y pata3, son las salidas D0 y D1, del puerto paralelo, correspondientes al bus de datos, pata 19 es una señal de tierra.

Por lo anterior se descartó la idea de utilizar este tipo de motores, que si bien fue fácil de manejar con la PC, también fue inconveniente, por su principal desventaja. Obviamente, se observó su gran utilidad para aplicaciones en robótica, en donde no se requiere de gran fuerza. Aunque si bien es cierto que existen motores de pasos de hasta varios caballos de fuerza aplicados en robots industriales, estos se vuelven caros, y de difícil adquisición en el país.

3.2.2 Motores de DC

Conociendo los resultados con el uso de motores de pasos, se pensó en utilizar algún motor de corriente directa de los conocidos como motorreductores, que no son más que un simple motor, cuya flecha está acoplada a un tren de engranes, que le resta velocidad pero gana en par.

Los motores empleados en el proyecto se obtuvieron en tiendas de partes electrónicas usadas, por lo que no fue posible tener a la mano sus especificaciones, por lo que empleando un método de inspección se observó que podrían fácilmente mover la carga que representa la parte mecánica. Estos motores pueden ser alimentados hasta con 18 V, el consumo máximo de corriente (para mover una carga similar al sistema

mecánico) se estimó en alrededor de 0 750 A. Por lo que el consumo máximo de corriente, cuando los dos motores están operando, es de 1.5 A.

Estos motores también son ampliamente utilizados en robótica, se toma en cuenta principalmente que su velocidad es proporcional a la tensión aplicada. Para los propósitos y condiciones de este proyecto, no es necesario el control de velocidad, lo anterior simplifica demasiado su control (el sistema de control es de los conocidos como ON - OFF). Es obvio cuestionarse el cómo se consigue el frenado en estos motores, cuando es absolutamente necesario que se detenga en forma precisa, para posicionar al sistema mecánico en las posiciones indicadas. Para contestar esto hay que recordar, se tiene un tren de engranes con un paso muy fino entre dientes, si recordamos que con el tren se pierde velocidad y se gana fuerza, se acepta que aunque el motor este apagado (después de estar energizado y de provocar movimiento), la flecha siga girando por efectos de la inercia. Esta fuerza alcanzara únicamente a mover algunos cuantos dientes de los engranes, sumado esto al peso del sistema, y a las fuerzas de fricción presentes, se traduce en una fuerza que causa el frenado casi instantáneo y con un error de posición insignificante (comparado con el área que tienen los micrófonos de la sonda, que cubren fácilmente el error que pueda ser obtenido de esta forma).

Teniendo seleccionado el tipo de motor a controlar, y a sabiendas que serian dos motores, y que estos tendrían que ser capaces de moverse en un sentido u otro. Se procedió al diseño de la tarjeta electrónica, capaz de hacer lo anterior.

3.2.3 Circuito de control para los motores

Con las ventajas antes mencionadas del motor seleccionado, se procedió a implementar un circuito de control para los motores. Este circuito recibe las ordenes de la PC, para que uno (ó dos) de los motores giren (en un sentido o en otro), mientras no se reciba otra orden que indique que se detenga ó no.

Estas ordenes se pasan, por medio del programa, al puerto paralelo quien se encarga de colocar señales eléctricas ó niveles de voltaje lógicos (1=5V ó 0=0V), en la tarjeta de control de los motores. Así que se eligió que con dos bits, fuese posible controlar a un motor y que con cuatro se controlara a dos motores.

Existen muchas alternativas para el control de motores de corriente directa, el control que se diseñó fue la opción más sencilla y rápida, ya que se simplifican mucho los cálculos y los circuitos.

El circuito de control diseñado se basa totalmente en la utilización del relevador de estado sólido conocido como SSR (solid-state relay). Estos relevadores son la contraparte de los relevadores electromecánicos EMR (electromechanical relay), su ejecución es la misma, se diferencian en que estos son dispositivos totalmente electrónicos que aprovechan diversas propiedades ópticas, eléctricas y magnéticas de los semiconductores. A parte en estos no existen los contactos mecánicos, que son los principales factores para que un EMR falle.

Los SSR proporcionan, un grado alto de confiabilidad, larga vida, reducen las interferencias electromagnéticas (EMI), respuesta rápida, no emiten ondas acústicas, son compatibles con los microcontroladores y tienen alta resistencia a las vibraciones. Sus desventajas principales son: costo, disipan mucha potencia (en algunos casos se requiere de disipadores de calor), son diseñados para AC o DC pero no para ambas, etc.²

En la figura 3.2.1 se muestra la arquitectura de un SSR, el bloque limitador de corriente sirve para proteger y evitar la destrucción del diodo emisor de luz (led), la luz del led se detecta por el fotodetector, que junto con el disparador (trigger) activan a los dispositivos de salida. Los dispositivos de salida pueden ser arreglos de transistores, scr's, triacs, diodos o combinaciones de estos. El circuito de protección, se encarga de proteger los dispositivos de salida del SSR, contra picos de corriente inversa, dV/dt , di/dt , etc.

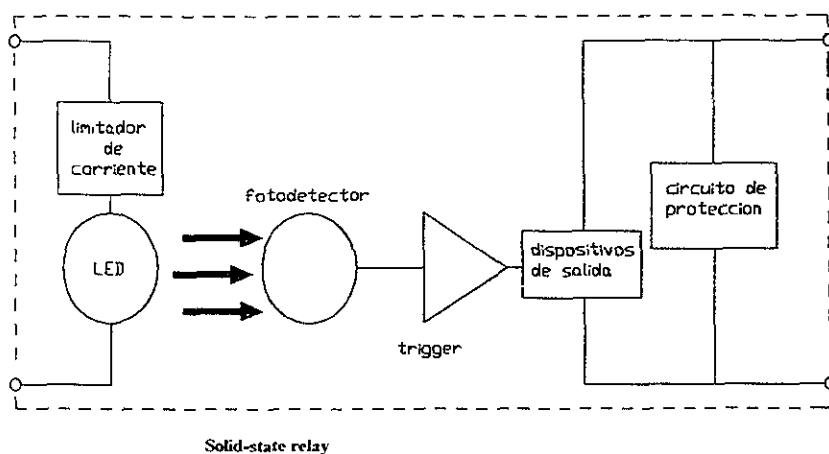


Figura 3.2.1 Diagrama a bloques de un relevador de estado sólido.

Los SSR's empleados en el proyecto son de la marca CRYDOM, en su serie para microcontroladores, el modelo es el DMP6101A. Fue elegido principalmente por sus siguientes características. entrada compatible con microprocesador o voltaje de dc, corriente de entrada de 32 mA, voltaje de entrada de hasta 32 V, corriente de salida de 3 A, tiempo máximo de encendido de 8.3 ms y tiempo máximo de apagado de 8.3 ms. Es recomendable consultar las hojas de especificaciones proporcionadas por CRYDOM company

En esencia el circuito de control se basa en el funcionamiento del circuito de la figura 3.2.2. Las entradas A y B, se encargan de controlar las tres acciones que puede tomar el motor, al aplicárseles los niveles lógicos correspondientes. Los circuitos lógicos se encargan únicamente de direccionar que SSR se activara. Si en A se tiene un nivel lógico 1, los SSR's estas capacitados para funcionar en el momento en que se les coloque un nivel lógico 0, este nivel lo proporciona la entrada B, cuando B tiene un nivel

alto, coloca un 1 en la compuerta NAND 1 y en el inversor 1, quien entrega un 0 al inversor 2, quien entrega un nivel alto, a la NAND 1, quien finalmente entrega un nivel bajo, al SSR1, quien cierra el circuito para que la fuente V+ mueva al motor en un sentido. Es obvio que en ningún momento se tienen que activar los dos SSR's al mismo tiempo, para esto se coloca la red RC que funciona como un retardo de tiempo que evita que en no menos de 8.3 ms, se conmute al segundo SSR, el tiempo de retardo de dicha red es de 10 ms, continuando con el análisis del inversor 2 se obtuvo un nivel alto que es entregado al inversor 3, quien en su salida da un nivel bajo, que aunado al nivel bajo del inversor1, hacen que la NAND2 de un nivel lógico alto, y en consecuencia el SSR2 este apagado. Un análisis similar se puede hacer cuando el nivel de B es bajo.

Como se menciono, la computadora, por medio de su puerto paralelo, coloca los diferentes niveles lógicos en A y B, siguiendo una secuencia dictada por un programa, afin de que el motor realice una acción específica.

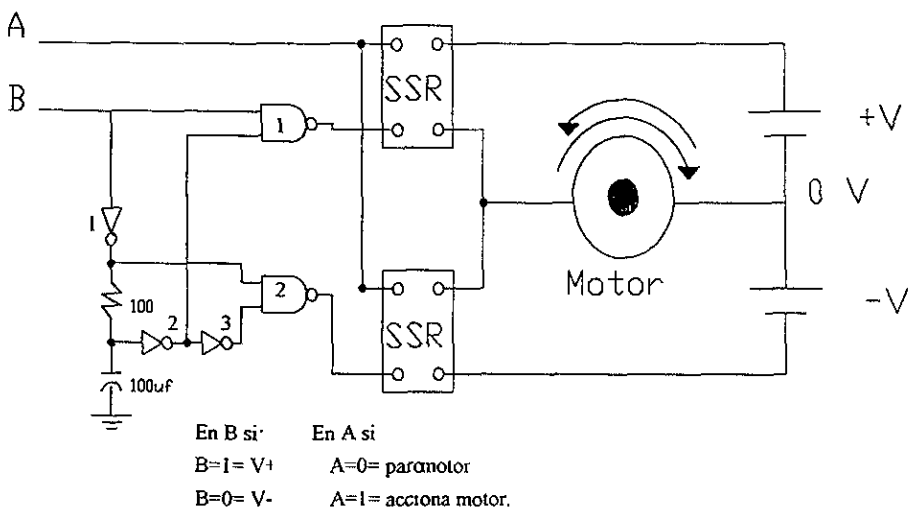


Figura 3.2.2 Esquema básico para el control de motores de DC, basado en los relevadores de estado sólido.

Para controlar dos motores simultáneamente y partiendo del funcionamiento del circuito de la figura 3.2.2, es necesario contar con cuatro señales lógicas de entrada que pueden llamarse como D0,D1,D2 y D3, cuyas combinaciones lógicas forman diferentes secuencias de funcionamiento, que ejecutarán ciertas acciones sobre los motores.

Con un nivel lógico alto en las entradas D0 o D2, se provee la capacidad, para que los motores puedan girar a la izquierda o a la derecha. Dicho giro se traduce en mover atrás, adelante, arriba y abajo, al sistema mecánico. Las entradas D1 y D3, controlan el sentido de giro.

La tabla 3.1, indica las combinaciones que pueden tener las cuatro entradas, y sus efectos en los motores *X* y *Y*, cuyo giro se traduce en el movimiento de las partes mecánicas, atrás, adelante, arriba ó abajo.

D3	D2	D1	D0	Motor X	Motor Y
0	0	0	0	paro	paro
0	0	0	1	atrás	paro
0	0	1	0	paro	paro
0	0	1	1	adelante	paro
0	1	0	0	paro	abajo
0	1	0	1	atrás	abajo
0	1	1	0	paro	abajo
0	1	1	1	adelante	abajo
1	0	0	0	paro	paro
1	0	0	1	atrás	paro
1	0	1	0	paro	paro
1	0	1	1	adelante	paro
1	1	0	0	paro	arriba
1	1	0	1	atrás	arriba
1	1	1	0	paro	arriba
1	1	1	1	adelante	arriba

Tabla 3.1 Secuencias para controlar a los dos motores.

Lo importante de esta tabla es que de ahí se derivan los diferentes niveles que se requieren para que el programa coloque en el puerto paralelo la secuencia ó palabra de control adecuada para efectuar alguna acción.

El puerto paralelo es la interfaz entre los programas y la electrónica capaz de convertir las señales digitales y proporcionar movimiento a la parte mecánica. Por lo pronto es necesario hablar sobre el circuito de control que ejecuta dicha acción y más adelante sobre los programas de control.

El circuito de control para los motores se puede ver en la figura 3.2.3, las señales D0, D1, D2 y D3 son señales provenientes del puerto paralelo, correspondientes a su bus de datos, C3* es una señal también del puerto paralelo pero correspondiente al bus de control. Estas señales se encargan de dar las ordenes al circuito de control para que el resultado final sea el movimiento de los motores en un sentido u en otro.

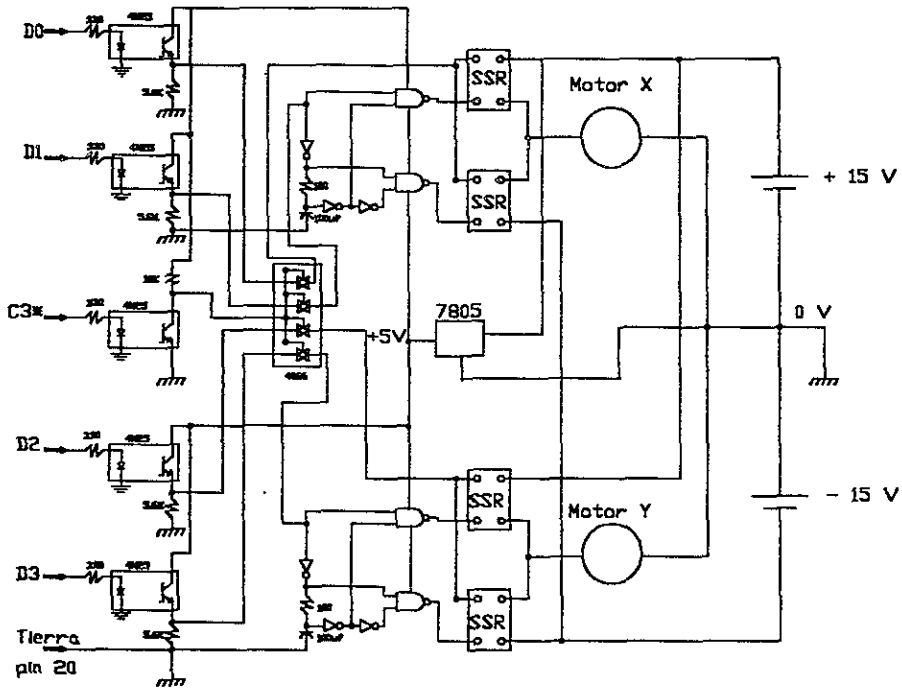


Figura 3.2.3 Circuito para el control de los motores

Como se observa, las señales del puerto paralelo están acopladas a este circuito por medio óptico, a fin de aislar esta etapa del puerto paralelo y evitar dañar a este último.

El acoplamiento se logra a través de circuitos optoacopladores con salida a transistor (se recomienda consultar hojas de especificaciones). Para las señales D0, D1, D2 y D3, se alimentan los leds, con resistencias limitadoras de corriente, y a sus respectivos fototransistores se conectan como compuertas no inversoras, es decir que si el transistor recibe un nivel alto, este entregara un nivel de voltaje alto por emisor. La señal C3* alimenta a un opto acoplador por medio de una resistencia limitadora de corriente que brinda, protección al puerto, el fototransistor se conecta como inversor, por que la señal de entrada es una señal negada (el asterisco indica la negación C3*)

Toda computadora personal cuando se energiza, inicia un proceso de arranque, que incluye, el colocar información en el puerto paralelo, esta información en un determinado momento haría que el sistema mecánico se moviera en forma no deseada, y lo peor de todo es que se moviera sin que el programa de control se esté ejecutando, para evitar este error de funcionamiento, se colocó un circuito integrado que contiene interruptores (switchs) analógicos, estos interruptores se conectaron a los optoacopladores que reciben a las señales D0, D1, D2 y D3. Los interruptores tienen una

entrada de control que cuando se le coloca un nivel lógico bajo, literalmente abre a los interruptores, y cuando se coloca control un nivel alto, los interruptores se cierran dejando pasar a las señales D0, D1, D2 y D3. El control de los interruptores proviene de la señal C3*, esta señal cuando se esta ejecutando el arranque de la computadora, adquiere un nivel lógico alto, que cuando sale del optoacoplador inversor, se torna en un nivel lógico bajo obligando a que los interruptores, se abran, consiguiendo con esto que no se activen los motores y en consecuencia que no se mueva el sistema mecánico. Unos segundos después la computadora termina su arranque limpiando el bus de datos (pone 0's), y colocando en un nivel bajo a la señal C3*. Cuando la señal C3* es baja el optoacoplador la invierte y coloca un nivel alto en el control de los interruptores, consiguiendo que los interruptores se cierren, permitiendo que las señales D0, D1, D2 y D3 pasen a los demás circuitos. Después del arranque los niveles de D0, D1, D2 y D3 es de 0 V, niveles que corresponden a que los motores este parados (ver tabla 3.1).

Después de que el circuito de control queda habilitado D0, D1, D2 y D3 entran a unos circuitos que tienen un funcionamiento idéntico al circuito de la figura 3.2.2 y cuyo funcionamiento ya se explico anteriormente. De ahí en adelante los movimientos que se requieren dependerán únicamente de los niveles que tomen las señales D0, D1, D2 y D3.

Hasta aquí se cuenta con un sistema mecánico movido por motores que son controlados con un circuito electrónico especial y por un programa de computadora, faltando por describir la etapa que identifica las posiciones a las que se quiere que el robot tome.

3.3 ORIENTACION

La orientación se refiere a la manera en que se logra proveer al sistema mecánico, la capacidad de conocer los puntos a que se quiere explorar, conociendo en que lugar esta y a cual tendrá que llegar. Y la forma en que se transmite esta información a la computadora, para que el programa decida que acciones se tomarán, colocando los niveles lógicos necesarios en las señales de control para los motores.

Para conseguir lo anterior es necesario colocar de alguna manera los dispositivos electrónicos que se encarguen de proporcionar información de la posición. Estos dispositivos electrónicos, conocidos como sensores, tienen que estar colocados de tal manera que no interfieran en el trayecto de las partes mecánicas, estos tienen que permitir el libre movimiento de los carros a lo largo de su respectivo eje, de un lado a otro, por lo que se pensó en sensores de tipo óptico y de tipo magnético.

Sumado a la selección del sensor, esta la forma en que se acondicionan estas señales, de tal manera que puedan ser utilizados en las tarjetas de interfaz entre los sensores y la computadora

3.3.1 Método para sensar la posición

Conociendo la estructura física de la parte mecánica y su funcionamiento, sumado a que el barrido es discreto, es lógico pensar en el empleo de sensores de tipo óptico o de tipo magnético, estos métodos de sensado no hacen necesario el contacto físico entre lo que se quiere detectar y el sensor en sí.

Para comprender lo anterior tenemos que ver la figura 3.3 1

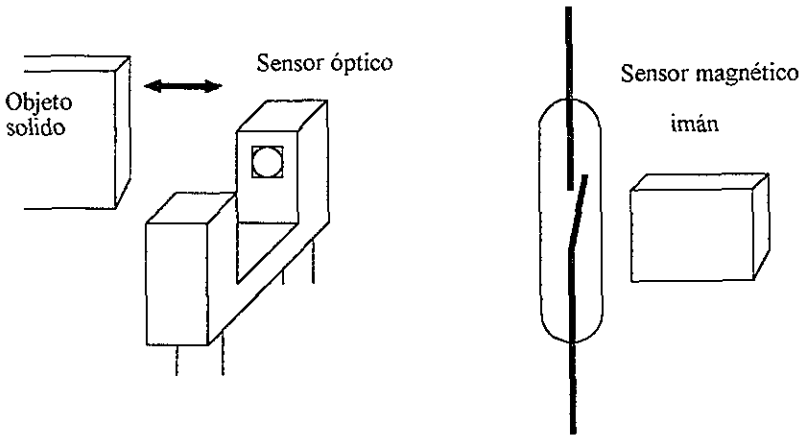


Figura 3.3.1 Sensores ópticos y magnéticos.

El sensor óptico está compuesto por un diodo emisor de luz infrarroja de los llamados led, y de un fotodetector que puede ser un transistor, una compuerta, o un disparador schmitt trigger. Se deben conectar circuitos de acondicionamiento de tal manera que detecten cuando un objeto sólido y obscuro interfiere entre el emisor y el receptor, bloqueando el haz de luz, los circuitos de acondicionamiento son instrumentados similarmente a los optoacopladores de la figura 3.2.3.

El sensor magnético, también conocido como reed switch (interruptor magnético), puede ser normalmente cerrado, normalmente abierto o tener ambas funciones, reaccionan cuando un flujo magnético los envuelve, su principal ventaja es que estos circuitos no requieren de polarización, por lo que en muchos casos su circuito de acondicionamiento resulta ser muy fácil y económico para su instrumentación.

Para la elección del mejor método se tomaron en cuenta, sus principales características, las demandas del sistema mecánico y los costos económicos. Anteriormente se habló que el barrido forma una rejilla o matriz de 16x16, por lo que es necesario el empleo de 32 sensores, y la consideración de los costos, tanto para los sensores como para los circuitos de acondicionamiento. La tabla siguiente ofrece una comparación entre estos sensores².

Sensor óptico	Sensor magnético
<ul style="list-style-type: none"> + No hay contacto físico entre el receptor y emisor. + Muy confiable para la detección. + Respuesta muy rápida. + Requiere de ser polarizado. + Son caros. + Necesita de dispositivos electrónicos adicionales para su acondicionamiento. 	<ul style="list-style-type: none"> + No hay contacto físico entre el receptor y emisor. + Muy confiable para la detección. + Respuesta rápida. + No necesariamente requiere polarización. + Son baratos. + Puede o no necesitar dispositivos electrónicos para acondicionamiento.

Es obvio que el método óptico resulta ser el más caro y un poco más difícil para su instrumentación, adicionalmente los sensores ópticos físicamente son más voluminosos que los sensores magnéticos, cosa que dificulta la instalación dentro del sistema mecánico.

Por las características anteriores se desechó el método óptico y se eligió el magnético, su única desventaja es su lentitud para reaccionar cuando un campo magnético lo envuelve, este defecto no resulta ser crítico, por que se corrigió por medio del programa dando retrasos de tiempo afin de que se pudiese detectar en forma adecuada la posición.

Los sensores magnéticos empleados tienen contactos normalmente abiertos, su encapsulado (de vidrio) contiene un gas especial que facilita la reacción y evita las chispas que afectan su vida útil, tienen una longitud de 1.5 cm y un diámetro de 0.15 cm.

Sobre los carros que se mueven dentro de cada eje se colocó un imán, respectivamente, así que cuando este imán pasa cerca de un sensor, cierra los contactos indicando al circuito que se encarga de la lectura de esta información, si se ha llegado a la posición deseada. En la figura 3.3.2 se muestra como se colocaron los sensores magnéticos, se puede ver que se perforó el riel para colocar las terminales de los sensores, obviamente el riel es de material conductor, por lo que las terminales de los sensores fueron aisladas. La distancia entre sensores es de 4 cm, así que para el eje X se cubre una distancia de 64 cm (16 sensores), para el eje Y tenemos, también, 16 sensores espaciados uno de otro a una distancia también de 4 cm por lo que se cubre una distancia de 64 cm. Aunque el tamaño de la rejilla que acepta el programa WT9378 de la marca Bruel & Kjaer es de 16 posiciones en X y 12 en Y (figura 3.1), por lo que en Y se tienen 4 posiciones de sobra, situación que no causa ningún inconveniente.

Para que el robot se mueva, por ejemplo a la posición (3,8), es necesario que los sensores colocados en la posición 3 del eje X y 8 del Y , se cierren al momento en que el campo magnético de los imanes montados en cada carro, los envuelvan, provocando que sus contactos se cierren.

Los imanes tienen una geometría cilíndrica (como se ve en la figura 3.3.2), pero bien pudiesen ser rectangulares ó cuadrados, lo importante es que tengan un campo magnético capaz de alterar el estado del sensor inmediato, sin alterar el estado de los sensores que tiene a los lados. La distancia entre la cara plana del imán y el sensor es de 0.5 cm, distancia que no altera los sensores contiguos, únicamente al sensor que tiene enfrente.

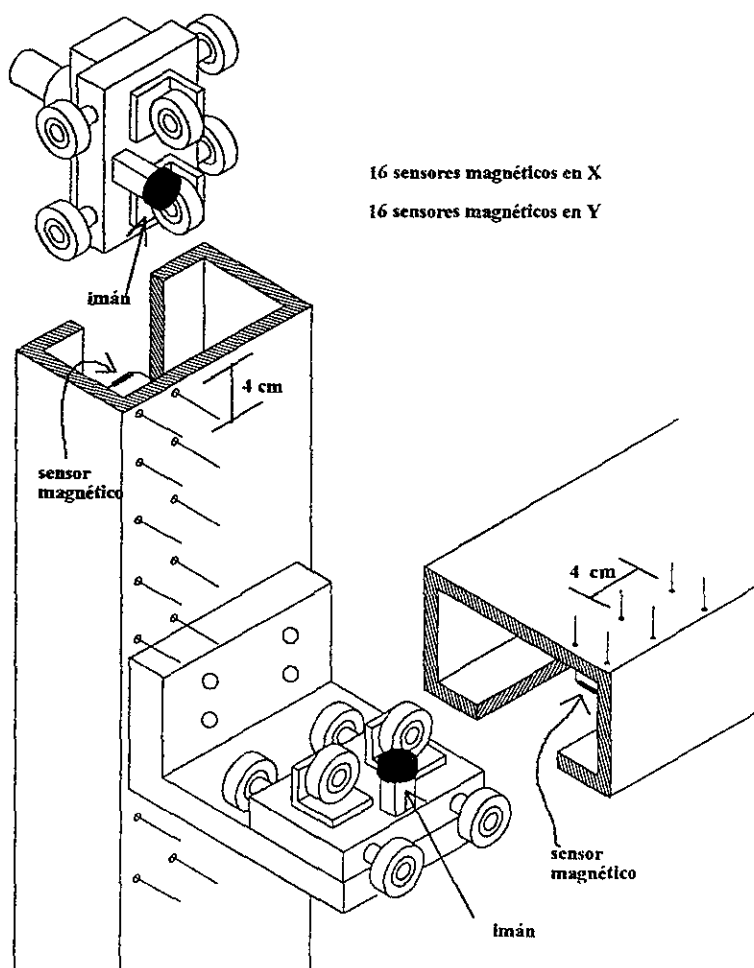


Figura 3.3.2 Ilustra la forma en que se colocaron los sensores en el sistema mecánico

Los circuitos que acondicionan la señal para que sea leída por la PC, se discutirán más adelante, por lo pronto se indicará la manera en que se conectan los sensores.

La conexión de los sensores es simple, únicamente se toma una de sus terminales como común, es decir que para el eje X se conectan 16 terminales a un solo punto, y las 16 terminales restantes entran al siguiente circuito. Lo mismo pasa para el eje Y . En la figura 3.3 3, se muestra la conexión que puede ser para el eje X y también para el Y

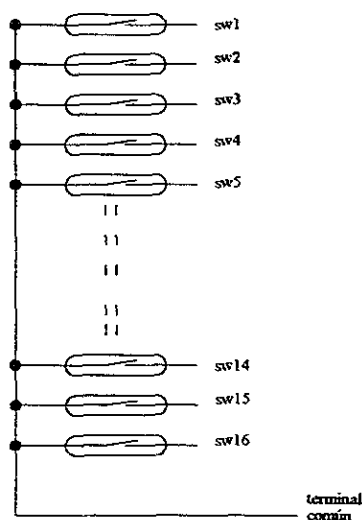


Figura 3.3.3 Se muestra como se conectan los interruptores magnéticos, para cualquiera de los ejes.

Para los 16 sensores del eje X y tomando como referencia el circuito de la figura 3.3.3, tenemos 17 terminales, así que también para el eje Y tendremos 17 terminales más, dando un total de 34 terminales ó cables que son enviados a la computadora. Para un manejo cómodo y principalmente para no interferir con las ondas sonoras a medir, la computadora se tiene que colocar a una distancia no menor de 3 m, distancia que multiplicada por 34 da un total de 102 m de cable, los cuales son voluminosos, son molestos para su manejo y también antiestéticos, por lo que se pensó en la forma de conocer las 16 posibles posiciones, por medio de dos hilos, dando un total de 4 hilos, para los dos ejes.

3.3.2 Sensado de la posición

En la figura 3.3.4 se muestra un diagrama a bloques de los circuitos que se encargan de efectuar el trabajo de sensar y reconocer la posición del robot. La computadora, mediante un programa puede leer la posición y a su vez controlar la tarjeta de expansión de puerto paralelo PCL-720, a dicha tarjeta se le conectan dos tarjetas hijas que permiten la conexión de 16 posiciones por cada eje. Las señales que leen estas tarjetas hijas (PCLD-782), provienen de los circuitos que detectan el estado de los sensores, por medio de dos cables o hilos.

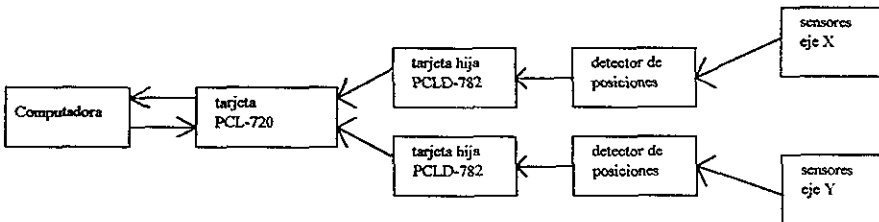


Figura 3.3.4 Diagrama a bloques, que muestra la forma en que se determina la posición del robot.

Las tarjetas PCL-720 y PCLD-782 (ADVANTECH), permitieron el diseño de un robot que puede mover la "cabeza" en 16 posiciones distintas para X y también 16 posiciones para Y , para así poder formar algo parecido a una matriz de 16×16 , equivalente a 256 posiciones distintas.

La tarjeta PCL-720, amplía el puerto entre otras cosas con³:

- * 32 canales de entrada digital.
- * 32 canales de salida digital
- * 3 canales con temporizadores programables.

Todos los canales de entrada y salida, de esta tarjeta son compatibles con lógica TTL/DTL. Cada entrada digital o salida digital, corresponde a un bit del puerto de I/O de la PC. El temporizador programable 8253, proporciona tres contadores de 16 bits. Adicionalmente la tarjeta contiene osciladores de 1 MHz, 100 kHz y 10 kHz, frecuencias que pueden ser duplicadas, divididas entre dos o entre cuatro.

Para facilitar las conexiones de la tarjeta PCL-720 se emplearon dos tarjetas PCLD-782, que proporciona 16 canales de entrada, cada una, aisladas ópticamente.

Estas tarjetas están equipadas con leds indicadores del nivel de entrada, si el nivel de voltaje en la entrada es alto el led esta prendido, las tarjetas se pueden trabajar en modo aislado (aisla ópticamente la señal de entrada del resto de la tarjeta), o modo no aislado. Se puede fijar el nivel de voltaje de disparo para las 16 entradas digitales.

(únicamente en modo aislado). Estas tarjetas se emplearon para monitorear los estadios que guardan 16 interruptores .

Dejando por lo pronto a un lado, los detalles de utilización de estas tarjetas, continuamos con la descripción del circuito detector de posiciones, el circuito sirve de interfaz entre los sensores del robot y las tarjetas hijas PCLD-782, pero a través de cuatro cables o hilos. Es obvio que se puede conectar directamente cada sensor en las tarjetas PC-LabCard, pero implicaría mandar del robot 34 cables ó hilos, por lo que se pensó en diseñar un circuito, que con dos hilos (por cada eje) mandara la información de la posición. El circuito que se encarga de lo anterior se realizó mediante un circuito integrado, de los empleados para señalizar el nivel de volumen en los equipos de audio. El LM3915 (consultar hojas de especificaciones de National Semiconductor). Ver figura 3.3.5.

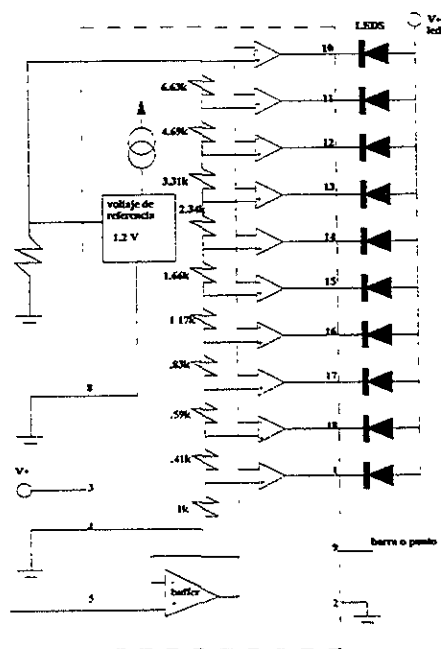


Figura 3.3.5 Diagrama a bloques del circuito integrado LM3915.

El circuito integrado LM3915, es un excitador barra de leds (que puede manejarse por barra o por punto), maneja hasta 10 leds indicadores. Básicamente se trata de un circuito que contiene 10 comparadores de voltaje con divisores de precisión, además contiene una referencia de voltaje ajustable. El circuito detector de posición, se basa en la comparación de 16 voltajes diferentes⁴, que son identificados por dos LM3915, conectados en cascada, así que cuando se le aplica un determinado voltaje (dentro de un rango de operación), alguno de los comparadores de los LM3915 conmuta a cero volts.

El LM3915 permite fijar diferentes niveles de voltaje en cada comparador. La señal de voltaje de entrada proviene de una red de resistencias a las cuales, en el momento de cerrarse algún interruptor (reed switch), hace posible la circulación de una corriente constante de 1mA (proporcionada por un circuito que satisface este fin y que aprovecha la referencia de voltaje del propio LM3915) En la figura 3.3.6, se tiene un circuito donde S1 a S16 son los 16 interruptores o sensores magnéticos, de un eje (para el otro eje el circuito es idéntico), y funciona de la siguiente manera. Debido a que el barrido es discreto nunca se tendrán dos interruptores cerrados al mismo tiempo, sobre el mismo eje, así que cuando se cierra el switch S_n donde n puede ser del 1 al 16, hará circular una corriente constante de 1mA por la resistencia R_n y se obtendrá un voltaje igual a $V_{sn} = 1 * R_n$ [mV].

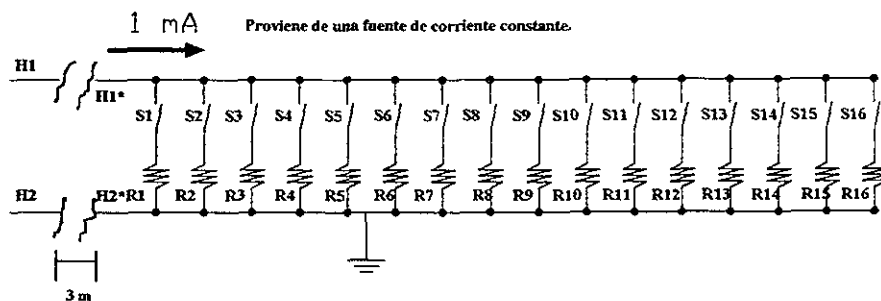


Figura 3.3.6 Circuito para proporcionar 16 diferentes voltajes.

El circuito de la figura 3.3.6 proporciona 16 voltajes diferentes que hacen que 16 comparadores diferentes conmuten, en el momento en que el sensor correspondiente es cerrado por la acción del campo magnético, proporcionado por el imán montado sobre los carros que están en movimiento. H1 y H2 únicamente indican las entradas del circuito en donde se encuentran los LM3915 (con el resto de dispositivos necesarios). Una longitud adecuada de cables entre H1 y H2 con H1* y H2* (terminales del circuito de sensores y resistencias), para un buen funcionamiento, sin que la impedancia de los cables sea significativa, es de 3m.

En la tabla 3.3.1 se pueden ver las 16 resistencias que forman el circuito de la figura 3.3.6, y los diferentes voltajes aproximados que proporciona este circuito

Resistencia	Voltaje
40 Ω	40 mv
50 Ω	50 mv
68 Ω	68 mv
88.7 Ω	88 mv
121 Ω	121 mv
182 Ω	182 mv
274 Ω	274 mv
392 Ω	392 mv

820 Ω	820 mv
1000 Ω	1 v
1500 Ω	1.5 v
2200 Ω	2.2 v
3300 Ω	3.3 v
4700 Ω	4.7 v
6700 Ω	6.7 v
8200 Ω	8.2 v

Tabla 3.3.1 Niveles de voltaje que son proporcionados por el circuito de la figura 3.3.6.

En la figura 3.3.7 se muestra el diagrama eléctrico del circuito que detecta los voltajes provenientes del circuito anterior, este circuito se montó sobre el mecanismo del robot. Así que, con dos hilos se manda un voltaje que entra al circuito de la figura 3.3.7, para que cuando este voltaje se compare, se active alguna de las 16 salidas (por eje), El CI1 tiene como voltaje de referencia 8 V, voltaje que es dividido internamente para fijar los diferentes niveles de comparación. De CI1 se utilizan las 8 primeras salidas y de CI2 se utilizan las salidas 2 a la 9, las cuales controlaran la activación de los optoacopladores de las tarjetas PCLD-782. Los primeros 8 voltajes de la tabla anterior, son amplificados, para que CI1 los compare y active la salida adecuada. A partir del noveno voltaje el CI1 ya no activara ninguna salida por que su voltaje de referencia habrá sido rebasado, El CI2 tiene una referencia de 9V, y comienza a funcionar a partir del noveno voltaje, correspondiente a 820 mV, la señal que entra a CI2 entra directamente de la red de resistencias, sin ser amplificada, por que llega con el suficiente nivel de voltaje para que lo detecte CI2, y para que CI1 no lo detecte, por que el factor de amplificación es de aproximadamente 12, entonces si para el noveno voltaje se tiene que $820 * 12 = 9.84 \text{ V}$, da un nivel que supera la referencia de CI1. CI1 tiene un rango de operación para sus primeros 8 comparadores de 350 mV - 3.9 V, obsérvese que el voltaje de entrada (ver los 8 primeros voltajes de la tabla 3.2.1) es amplificado 12 veces. CI2 tiene un rango de operación para sus comparadores, correspondientes a las salidas 2 a 9, entre 598 mV - 6.78 V.

Internamente los circuitos integrados LM3915 tienen un voltaje de referencia de 1.2 V que se utilizó para proporcionar una corriente constante de 1 mA, dicha fuente de corriente se obtiene a través de un transistor.

Un detalle que hay que cuidar es el diseño del circuito impreso, con respecto a las tierras, afin de no tener impedancias parásitas que afecten el buen funcionamiento del circuito

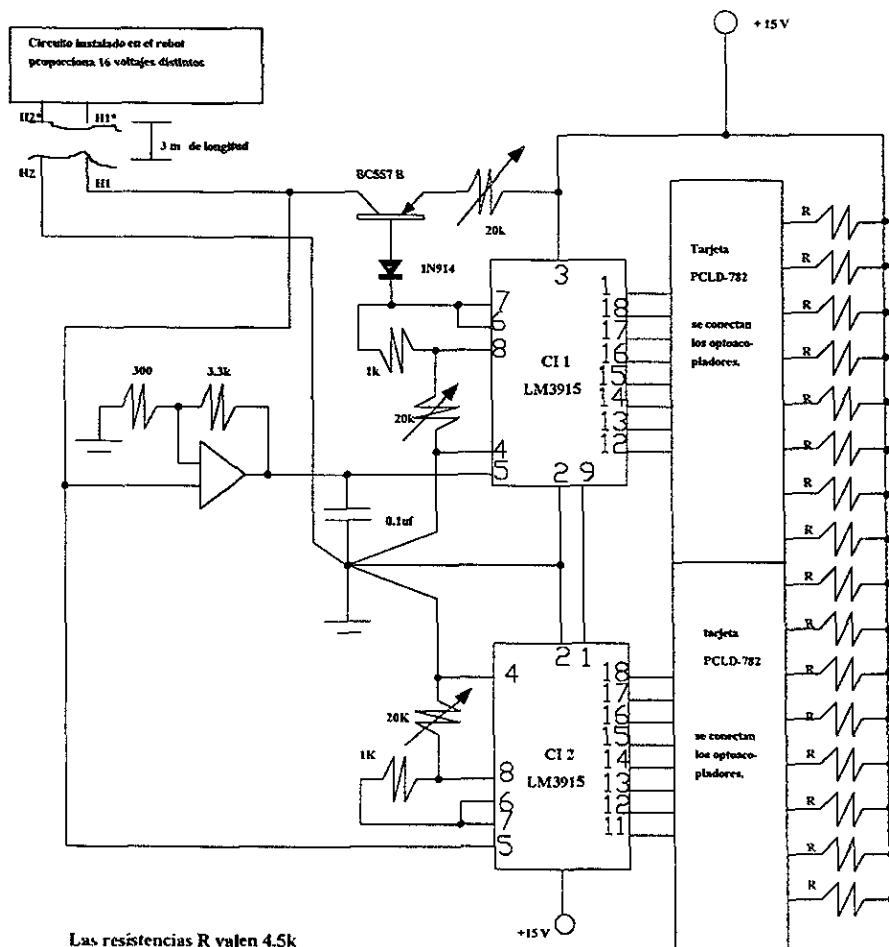


Figura 3.3.7 Circuito de interfaz entre las tarjetas PC-LabCards y el robot.

Los circuitos de las figuras 3.3.6 y 3.3.7, constituyen la interfaz (que bien puede llamarse control remoto de dos hilos para 16 interruptores) que interconecta al robot con la computadora apoyándose en otras tarjetas, como son la PCL-720 y la PCLD-782, de ésta última tarjeta se muestra su circuito en la figura 3.3.8

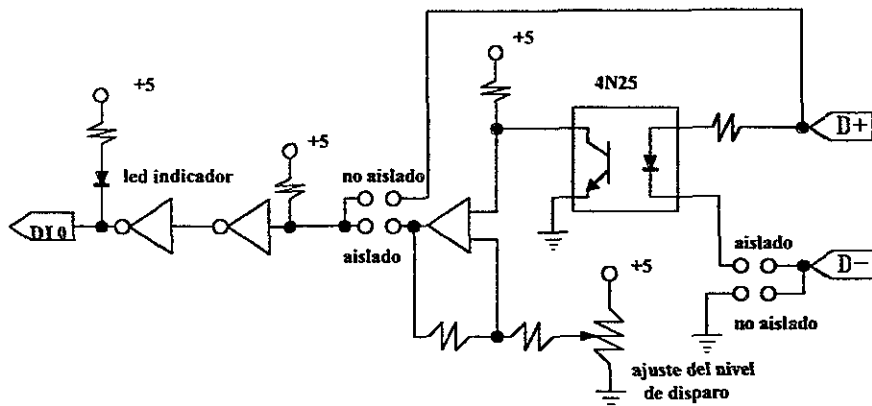


Figura 3.3.8 Diagrama de un canal digital de la tarjeta PCL-782, hay que recordar que son 16 canales⁵.

Para conectar las salidas de los LM3915 a las entradas de las PCLD-782, primero que nada, se seleccionó (por medio de los respectivos jumpers) el modo de trabajo aislado, con la finalidad de utilizar el comparador y poder ajustar el voltaje de referencia para la conmutación del comparador y evitar con esto posibles disparos en falso. En las entradas D+ y D- se conectan las salidas del circuito detector de posiciones, en la entrada D+ se conecta una resistencia de 4.7 k quien a su vez esta conectada a la fuente de 15 V, esta resistencia es para limitar la corriente de entrada al led del optoacoplador, la entrada D-, se conecta directamente a una de las salidas del LM3915, recordando que lo anterior es idéntico para los 32 canales 16 del eje X y 16 del Y, ahora, las señales de salida de estas tarjetas se conectan a la tarjeta PCL-720, quien recibe los 32 canales de entrada digital

La conexión de las tarjetas PCLD-782 con la PCL-720, se efectúa a través de conectores de 20 patas y de cables planos.

La instalación de la tarjeta PCL-720 dentro de la computadora, se hace únicamente insertando la tarjeta dentro de una ranura de expansión del bus interno de la PC. Asta aquí queda conformado el medio electrónico para leer posiciones, y que se ilustra en la figura 3.3.4.

Teniendo el medio electrónico de adquisición de la posición es necesario conocer, esta información. La computadora es el medio que se encarga de leer esta información. La información la obtendrá en forma directa de la tarjeta PCL-720.

A la tarjeta PCL-720, se le asigna una dirección dentro del espacio de direcciones del bus de la PC, para este proyecto se utilizó la de omisión 2A0 (hex). Esta tarjeta tiene 8 registros de direcciones por lo que ocupa un espacio que va de la dirección 2A0 - 2A7.

La siguiente tabla muestra las locaciones de cada registro, D/O significa Digital Output, D/I Digital Input, LSB byte menos significativo, MSB byte más significativo.

	Escritura	Lectura
2A0+0	D/O 0-7	D/I 0-7
2A0+1	D/O 8-15	D/I 8-15
2A0+2	D/O 16-23	D/I 16-23
2A0+3	D/O 24-31	D/I 24-31
2A0+4	LSB o MSB del CONTADOR 0	
2A0+5	LSB o MSB del CONTADOR 1	
2A0+6	LSB o MSB del CONTADOR 2	
2A0+7	Byte de control	

Para leer, por ejemplo las entradas de 0-7, con lenguaje C, basta con utilizar la siguiente línea `_inp(2A0+0)` la instrucción indica la lectura de la información que contiene la dirección que esta entre paréntesis⁶. Dicha operación entrega la lectura de los bits 0-7, de este byte correspondiente a los canales 0-7.

Ya mencionamos que por cada eje se ocupa una tarjeta hija PCLD-782, físicamente cada tarjeta PCLD-782 recibe en dos paquetes de 8 bits⁵, sus 16 canales de entrada, es decir que dos byte's contienen la información del estado de las entradas, como se ocupan dos de estas tarjetas, resulta que en cuatro bytes estará contenida la información del estado de las entradas que dependen de la posición de la cabeza del robot. Así que para conocer en que posición esta la cabeza, o mejor dicho en que posición el robot coloca la sonda, únicamente hay que leer el contenido del registro adecuado. La forma en que se logra lo anterior se menciona en la siguiente sección.

Sabemos ya que son cuatro registros los que contienen la información de la posición, por lo que el plano X - Y se dividió en cuatro, los canales 0-7 y 8-15 corresponden las posiciones de la 1-16 del eje X , para los canales 16-23 y 24-31 corresponden las posiciones de la 1-16 del eje Y . Así que el plano X - Y , quedo dividido en cuadrantes, compuestos por los segmentos (a,c), (a,d), (b,c) y (b,d), donde a y b, son los dos segmentos en que esta dividido el eje X , c y d son los segmentos del eje Y . En la figura 3.3.9 se puede ver lo anterior.

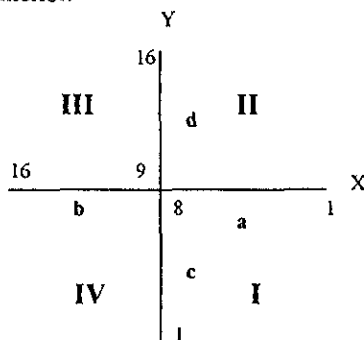


Figura 3.3.9 Distribución de los canales de entrada digital, proporcionados por la tarjeta PCL-720

El segmento **a** contiene el byte de información correspondiente a la dirección 2A0+0, el **b** al correspondiente a la dirección 2A0+1, el **c** el byte correspondiente a la

2A0+2 y el d el byte correspondiente a la dirección 2A0+3, a es el byte con la información de la posición 1-8 del eje X , b es el byte con la información de la posición 8-16 también del eje X , c es el byte con la información de la posición 1-8 del eje Y , y por último, d es el byte con la información de la posición 9-16 del eje Y .

Cada posición que toma el robot entrega un byte diferente (tomando en cuenta que cada tarjeta esta dividida en dos byte's con sus respectivas direcciones), como sabemos un byte esta compuesto de 8 bits, entonces cada bit de los byte's que entregan las tarjetas, indicara la posición del robot mediante el estado lógico que guarden sus bits, si el nivel lógico que se lee en algún bit es alto indicara que el robot a colocado a la sonda en esa posición. Debido a que son 8 bits se tiene que solo son ocho valores diferentes los que tomara el byte por cada segmento, para leer estos valores únicamente hay que indicar la dirección correcta mediante el programa.

Para el eje X las 16 posiciones se obtienen de la tabla 3.3.2, si la dirección es la 2A0+1 y el dato es 9, por ejemplo, quiere decir que el valor en hexadecimal es el 01, correspondiente a los interruptores magnéticos de la posición 9 del eje X se ha cerrado.

2A0+1 b	2A0+0 a	Dato en hexa.	Dato binario
9	1	01	00000001
10	2	02	00000010
11	3	04	00000100
12	4	08	00001000
13	5	10	00010000
14	6	20	00100000
15	7	40	01000000
16	8	80	10000000

Tabla 3.3.2

2A0+3 d	2A0+2 c	Dato en hexa.	Dato binario
9	1	01	00000001
10	2	02	00000010
11	3	04	00000100
12	4	08	00001000
13	5	10	00010000
14	6	20	00100000
15	7	40	01000000
16	8	80	10000000

Tabla 3.3.3

Para el eje Y se tiene una tabla 3.3.3 que es similar únicamente varia en las direcciones, si la dirección es 2A0+2 y el interruptor magnético que se cerró es el 4 correspondiente a la posición 4, el dato en hexadecimal será el 08, si juntamos el ejemplo para el eje X con este ultimo, da la posición (9,4).

En general la información contenida en las dos tablas anteriores, indica el valor en base hexadecimal, correspondientes a las posiciones, y que el programa se encargará de

leer, para que con esto se tome alguna decisión. El programa sabe qué direcciones de las tarjetas Pc-Lab leer.

Resumiendo, la operación de adquisición de la posición, se efectúa de la siguiente manera: cuando un interruptor magnético se cierra se aplica una corriente constante de 1mA, a una determinada resistencia que por ley de ohm da también un determinado voltaje, el cual pasa a través de dos hilos al circuito detector de posición, aquí se compara este voltaje, resultado de esta última operación es la excitación del emisor de un optoacoplador específico, dentro de la tarjeta PCLD-782, esta tarjeta coloca también la información en la tarjeta PCL-720, quien a su vez deja disponible la información dentro del bus interno de la PC, para su lectura mediante un programa que de acceso a la dirección en que esta la información.

3.4 PROGRAMAS

Hasta aquí se han descrito los elementos que conforman al robot, y que son: la parte mecánica, los circuitos de control de los motores y los circuitos de adquisición de la posición.

El elemento con que se logró controlar a estos elementos, es una computadora, mediante programas. La interfaz entre la computadora (PC) y los circuitos electrónicos de control es el puerto paralelo.

Dentro del espacio de direcciones para puertos solo se ocupa la dirección correspondiente al puerto paralelo 1 ó LPT1. Las líneas de I/O del puerto paralelo, que comúnmente también se le llama puerto para impresora, tienen las siguientes direcciones correspondientes, a los registros con nombre⁷:

<i>Nombre</i>	<i>Dirección para LPT1</i>	<i>Tipo de función</i>
Puerto de datos	378h ó bien 888d	Input/Output
Puerto de estado	379h ó bien 889d	Input
Puerto de control	37Ah ó bien 890d	Output

Las direcciones que tienen escrito al final una "h" significa que es base hexadecimal y una "d" significa que es base diez. El conector del puerto paralelo es del tipo DB25 y en él se hallan todas las líneas correspondientes al puerto y sus funciones se muestran en la tabla de la siguiente página.

DB25 pin #	Registro	Función	Bit
1	Control	Output	C0*
2 - 9	Datos	Input/output	D0 a D7
10	Estado	Input	S6
11	Estado	Input	S7*
12	Estado	Input	S5
13	Estado	Input	S4
14	Control	Output	C1*
15	Estado	Input	S3
16	Control	Output	C2
17	Control	Output	C3*
18 - 25		Tierra	

Si recordamos la tabla 3.1, se puede ver que se necesitan cuatro señales para controlar las acciones de los dos motores, estas señales son proporcionadas por los bits D0, D1, D2 y D3 del puerto paralelo correspondientes a las patas 2, 3, 4 y 5 del conector DB25, de la pata 17 se obtiene la señal C3*, cuya utilización ya se explico.

Antes de continuar, recordaremos que la PC se ayuda también de la tarjeta PCL-720, que entre otras cosas contiene al temporizador programable 8253, que ocupa 4 locaciones en el espacio de direcciones de I/O. Estas direcciones se muestran a continuación.

	Escritura	Lectura
2A0+4	LSB o MSB del CONTADOR 0	
2A0+5	LSB o MSB del CONTADOR 1	
2A0+6	LSB o MSB del CONTADOR 2	
2A0+7	Byte de control	

Se menciona lo anterior por que se utilizó el contador 0 del 8253, en modo de monoestable one-shot, para detectar algún error en el robot, ampleandolo de la siguiente forma, cuando el robot se mueve de una posición a otra posición deseada transcurre un determinado lapso de tiempo que no debe ser mayor al tiempo definido por el monoestable, si el robot alcanza la posición dentro del tiempo programado en el monoestable el programa entrega un código al sistema para que los demás programas acepten que se ha conseguido posicionar al robot en el lugar correcto y continuar así con la elaboración de los mapas, si por el contrario, el robot nunca alcanzó la posición dentro del lapso de tiempo, se ordena mediante programa, que se detengan los motores, colocando dentro del sistema un código de error, que detendrá todo proceso hasta que se detecte la posible falla. Para programar al 8253 se tiene que colocar un determinado dato en el registro del byte de control, así que para los fines del proyecto se coloco el dato 32 en hexadecimal en el byte de control, cuyo formato es el siguiente:

Bit	7	6	5	4	3	2	1	0
32h	SC1	SC0	RL1	RL0	M2	M1	M0	BCD
32h	0	0	1	1	0	0	1	0

Con el byte de control conteniendo el dato 32h o 50d se programó el

funcionamiento del contador; la frecuencia de reloj utilizada fue de 10KHz / 4, es decir de 2.5 KHz, o bien 0.4 mseg.

El disparo del monoestable lo efectúa la señal C0* (pata 1 DB25) del puerto paralelo, al generarse un flanco positivo (por medio del programa) en este bus o registro, esta señal se conecta en la pata 12 del conector 5 de la tarjeta PCL-720 (consultar manual), que es la pata del GATE 0 (pata donde se recibe un flanco positivo para el disparo del monoestable). La lectura del nivel lógico del monoestable se efectúa por medio del registro o bus de estado, a través de S6 (pata 10 DB25) quien esta conectada en la pata 10 OUT 0 del conector 5 (pata de salida del monoestable) de la PCL-720. El registro de estado lee un nivel 0 cuando el monoestable se va abajo a la hora del disparo y cuando ha cumplido el tiempo programado la salida toma un nivel alto.

Los programas, se encargan de manejar a todas las tarjetas de control del robot, coordinando todas las acciones necesarias. Son dos programas de comando del robot, el primero sirve para que el robot coloque la sonda en la posición inicial (1,1), ya que es necesario colocarla siempre que se va a iniciar un barrido, en la posición inicial, con el fin de tener un punto de referencia y evitar que el robot pierda su ubicación, enseguida se muestra el programa que se encarga de realizar esta función su nombre es SETROB.C.

El otro programa entra en operación después de que se tiene la sonda en la posición inicial (1,1), sirve para mover en cualquier dirección y en cualquier cuadrante a la sonda, se puede mover en las cuatro direcciones dentro de un cuadrante ó bien se puede desplazar de un cuadrante a otro. como se pueden ver marcados como I, II, III, IV en la figura 3.2.9, el nombre de este programa es ROBOT.C.

Los listados del programa aparecen en el apéndice A. Finalmente se da una breve descripción del programa ROBOT.C. primero se conoce el destino al que se tendrá que mover la sonda, y que son las coordenadas (x,y), el programa identifica entonces el cuadrante en el que se ubica dicha posición, a su vez se reconoce la posición actual ó posición activa en que esta la sonda, los datos recabados son pasados a las funciones cuadrante_1, cuadrante_2, cuadrante_3 y cuadrante_4, en estas funciones se detecta la posición activa y se le asigna un numero entero a cada coordenada, del 1 al 16, hecho esto y conociendo la posición deseada, se decide que acción tomar, es decir se elige accion_1, accion_2, accion_3, accion_4, estas funciones activan los motores en sentidos correctos con el fin de alcanzar la nueva posición, además de checar las tarjetas PC-Lab, y al temporizador.

Estos dos programas son como ya se había mencionado, hechos en lenguaje C ver. 7 de Microsoft. Los cuales se incrustaron dentro de un programa en lenguaje Pascal, el cual también esta contenido dentro del programa comercial WT9378, programa que se encarga de coordinar al analizador de frecuencias y al robot para tomar las lecturas en cada punto de medición. Más adelante se hablara y se mostrara la manera en que se hizo esto.

Asta este punto se ha detallado la teoría de la intensidad acústica y el método de medición empleado, además se ha detallado el diseño del robot, ahora pasaremos a detallar como se consigue tomar los datos sobre una rejilla, guardarlos y procesarlos, con la PC. Para esto necesitamos los medios que sirvan de enlace entre los "sistemas de adquisición y procesamiento de datos" y el " sistema de posicionamiento y orientación", con la PC.

REFERENCIAS CAPITULO 3

1. K. Takashi, A. Sugawara, Stepping motors and their microprocessors controls, (Ed. Clarendon), (1994).
2. Crydom, Solid Stated Relay Handbook(1990)
3. C-LabCards, User's Manual Digital I/O & Counter Card Model PCL-720,(1988)
4. A. Birkett, Electronics World Jun, 502, 1997
5. PC-LabCard, PCLD-782 16 Channel Opto-isolated D/I Board
6. Microsoft Corporation, Run-Time Library Reference Microsoft C/C++ Version 7.0
7. D.V. Gadre, K. Pramod, V. S. Varma, Computers in Physics, vol. 8, (1989)

CAPITULO 4

4. SISTEMA DE ADQUISICION Y DE PROCESAMIENTO DE DATOS

El sistema de adquisición y procesamiento de datos, se encarga de recopilar la información de tipo acústico que emiten las fuentes sonoras, y de procesar dicha información a fin de entregar los mapas de intensidad sonora ya sea en papel o en una gráfica en la computadora. Este sistema esta integrado, además de la computadora, por los siguientes elementos:

- 1.- Tarjeta de interfaz IEEE-488 (NI-488.2 de National Instruments).
- 2.- Analizador de frecuencias 2133 marca B&K.
- 3.- Sonda de intensidad sonora 3545 marca B&K, con su control remoto.
- 4.- Programas de control

La sonda de intensidad sonora se encarga de transformar la información acústica en información eléctrica que después se procesa por el analizador B&K 2133. Posteriormente, esta información se envía a un buffer de memoria en la computadora (lugar en donde se almacenan los datos de intensidad para todo un barrido de la sonda).

Para que dicha información pueda ser recibida en la computadora se emplea una interfaz entre el analizador de frecuencias y la computadora, dicha interfaz es la IEEE-488 (tarjetas que también son conocidas como GPIB).

Para que este sistema funcione es necesario contar con los programas de control, para esto se utilizó:

- a) un programa comercial WT9378 marca Brüel & Kjær
- b) los programas de posicionamiento del sistema mecánico (o bien, sistema de posicionamiento y de orientación de la sonda o robot).
- c) el programa de análisis de la información hecho en Matlab.

El programa WT9378 se encarga de la adquisición de datos y los controles del analizador B&K 2133. Este programa cede el mando al programa de control del sistema de posición y orientación, que mueve las partes mecánicas a fin de colocar a la sonda en el punto adecuado para la medición. Después, el programa WT9378 retoma el control para que el analizador efectúe la medición, se transfiere este dato al disco duro de la computadora y vuelve a ceder el mando al programa que controla la parte mecánica para enviarla a la siguiente posición en que la sonda realizará la siguiente medición. Finalmente, cuando la sonda ya entrego todos los datos del barrido indicado y estos están en un archivo en el disco duro, es posible visualizar el mapa de intensidad sonora, en forma gráfica ya sea en papel ó en pantalla, por medio del propio programa WT9378 o por un programa de análisis adicional hecho en Matlab

4.1 TARJETA DE INTERFAZ (PROTOCOLO DE COMUNICACION IEEE-488)

Como se menciona en la introducción de este capítulo y para que el proyecto cumpla con la automatización del proceso de elaboración de mapas de intensidad sonora, es necesario que el sistema funcione con las tecnologías actuales que dan facilidades y permiten mayor versatilidad. Una de las facilidades que nos dan los instrumentos actuales, y en general cualquier equipo sea este para instrumentación ó no, es la capacidad de poder ser conectados a una computadora personal con el fin de lograr integrar un equipo de mayor capacidad en cuanto a la adquisición de información y procesamiento de la misma. Para dar ejemplos, actualmente existen osciloscopios con interfaz para computadora, calculadoras, multimetros, etc. .

El protocolo de comunicación de la interfaz IEEE-488 ó GPIB-488 fue diseñado por la corporación Hewlett Packard, que fue la primera en diseñar una interfaz para instrumentación en 1972 y subsecuentemente se desarrolló hasta lograr el protocolo HP-IB, que tuvo mucha aceptación por sus amplias capacidades en el manejo de dispositivos y por que permitía crear sistemas de instrumentación flexibles. El protocolo HP-IB fue adoptado por el Instituto de Ingenieros Eléctricos y Electrónicos (IEEE) como norma (IEEE-488) en el año de 1975, también por la ANSI MC1-1975 denominándola como, bus de interconexión de propósito general, cuyas siglas en ingles son GPIB (General Purpose Interface Bus), esto presumiblemente por la negativa de las otras compañías de usar las siglas de HP. Finalmente la norma fue revisada en 1978 para incorporarle cambios menores. Cabe mencionar que la patente la posee Hewlett Packard, pero la compañía esta comprometida con la IEEE para ceder la licencia de utilización a quien lo solicite, en términos razonables¹.

La marca National Instruments, tiene una tarjeta que es compatible con la interfaz IEEE-488, cuyo modelo comercial es la NI-488.2.

En el apéndice B se presentan los detalles de la IEEE-488, y en el apéndice C las características de la tarjeta NI - 488 2 que se utilizó en este proyecto

4.2 ANALIZADOR DE FRECUENCIAS 2133 DE B&K

El analizador B&K 2133 es un analizador de frecuencia por filtrado digital, para señales de sonido y vibraciones, en bandas seleccionables desde 1/1 hasta 1/24 de octavas y con operación en tiempo real en un ancho de banda de 22.4 kHz (en bandas 1/3 de oct.).

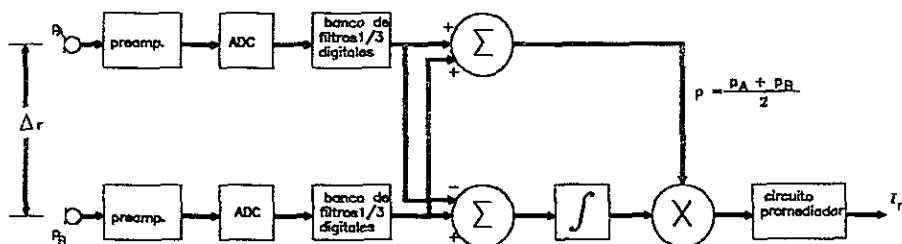
El analizador procesa la información en tiempo real, por medio de funciones matemáticas predeterminadas, en memoria no volátil, que evalúa el procesador digital de señales (DSP). Dentro de estas funciones, están las correspondientes para el cálculo de la intensidad sonora activa y la intensidad sonora reactiva, las cuales están definidas como²:

Intensidad sonora activa.

$$\frac{P_A + P_B}{2\rho\Delta r} \int (p_A - p_B) dt$$

Intensidad sonora reactiva:
$$\frac{(p_A + p_B)(p_A - p_B)}{2\omega\rho\Delta r}$$

La intensidad activa se refiere a la porción de energía que se propaga y la intensidad reactiva, es la porción de energía sonora que no se propaga. Ya se ha mencionado, anteriormente, que las mediciones hechas en este proyecto son de la intensidad sonora activa. La función que hace el analizador se puede visualizar en la siguiente figura.



Las señales de los dos micrófonos entran a cada uno de los canales del analizador respectivamente, estas señales son convertidas a información digital por medio de los ADC's, después la señal digital entra al banco de filtros también digitales, posteriormente se obtiene la diferencia de las señales siendo entregadas al bloque integrador, ahora su salida va al multiplicador, la otra entrada del multiplicador es la suma de las dos señales; este ultimo bloque entrega la señal:

$$(p_A + p_B) \int (p_B - p_A) dt$$

Una señal a la cual solo le falta ser multiplicada por el factor.

$$\frac{-1}{2\rho\Delta r}$$

Para que al final se obtenga la intensidad sonora activa, o bien la reactiva que tiene un proceso similar para ser calculada por el analizador.

El analizador "2133" tiene una entrada especial que permite la conexión con el control remoto ZH0354 (que a su vez, conecta a la sonda). La comunicación entre el analizador 2133 con la computadora se lleva a cabo con la interfaz IEEE-488. El programa WT9378, se encarga de controlar, en modo remoto, al analizador, programándolo en el modo de medición (setup), adecuado para las mediciones de intensidad (según el manual de usuario del analizador, setup numero 12). Es importante señalar que se pueden controlar en forma remota casi todas las funciones del analizador a excepción de la polarización, el aterrizaje y algunos parámetros como son: la temperatura ambiente, la presión atmosférica, y las sensibilidades de los micrófonos. Por lo que se tienen que efectuar las actividades anteriores manualmente.

4.3 SONDA DE INTENSIDAD SONORA TIPO 3545

El dispositivo que se encarga de obtener la información acústica es la sonda o sensor de intensidad sonora tipo 3545. Esta sonda esta compuesta por un par de micrófonos tipo 4181 y otro par del tipo 4178, ambos de B&K, emparejados en fase y amplitud. El principio de transducción es del tipo 'p-p' con una configuración cara a cara y espaciados por un separador sólido. Como ya se menciono este tipo de sonda utiliza un método llamado de aproximación de diferencias finitas, la cual aproxima el gradiente de presión a la diferencia de presiones que captan los dos micrófonos de la sonda, y dividida entre la distancia que los separa.

El que los micrófonos de la sonda estén emparejados, significa que deben de tener una respuesta dinámica similar. La figura 4.3.1 ilustra esto. Dos micrófonos son excitados por una onda plana en una cámara anecoica. Están puestos cara a cara con un separador sólido de 12mm, ambos deben de registrar la misma presión, el separador sólido de los micrófonos, disminuye la difracción al rededor de los bordes de las rejas de los micrófonos. De esta forma se obtienen respuestas con diferencias de 1dB.

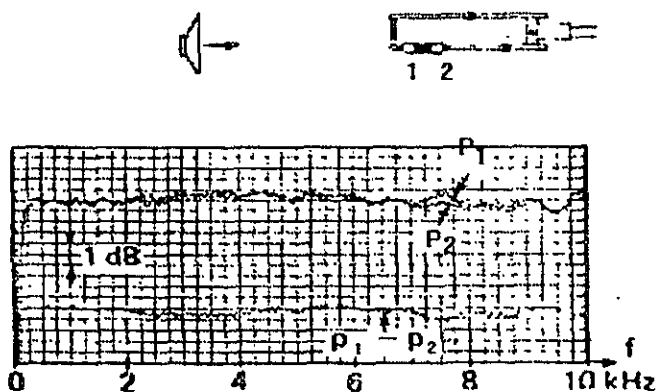


Figura 4.3.1 Respuesta real de dos micrófonos nominalmente idénticos, en sondas con configuración cara a cara, con espaciador sólido.

El espaciador ó separador sólido es muy importante, ya que del depende si una medida será realmente aceptable. Una limitación que puede causar un error en la medición, por el método de aproximación de diferencia finita, es el que existen limites de respuesta en la frecuencia para que una lectura sea aceptable, así que el limite para alta frecuencia, o error de alta frecuencia es también llamado error de *bias* (error de tendencia) que da una falsa estimación de la intensidad que se mide. La sonda como ya sabemos, aproxima el gradiente de presión de una curva, a una línea recta entre dos

puntos. Si los cambios de la curva son demasiados rápidos con respecto a la distancia, la estimación será por lo tanto inadecuada. Esto pasará si la longitud de onda medida, es pequeña comparada con la separación efectiva de los micrófonos, ver figura 2.3.4

Para cualquier separación efectiva entre los micrófonos siempre habrá un límite en alta frecuencia que hará que los errores aumenten significativamente. Para que una lectura tenga una exactitud entre 1dB, la longitud de onda medida tiene que ser seis veces más grande que la longitud del separador. Que corresponde a los siguientes límites en alta frecuencia³:

<i>50mm:</i>	<i>hasta 1.25 KHz</i>
<i>12mm:</i>	<i>hasta 5 KHz</i>
<i>6mm:</i>	<i>hasta 10 KHz</i>

Si la fuente sonora esta colocada como en la figura 4.2.1 (a), la intensidad en esta dirección es considerada como positiva, para el caso contrario figura 4.2.1 (b), las ondas se desplazan por la parte trasera de la sonda, la intensidad es considerada negativa



Figura 4.2.1 Dirección de la sonda para las mediciones.

4.4 PROGRAMA DE CONTROL WT9378

En esta sección se explica como se controlan los aparatos de medición de la intensidad acústica por medio de la computadora (PC), el medio que se encarga es el programa WT9378.

Este programa se encuentra en código ejecutable bajo el sistema operativo MS-DOS. El programa sirve para el mapeo de potencia sonora, obviamente para mapear potencia sonora, es necesario que procese primero la intensidad sonora, para esto el programa puede controlar una amplia gama de analizadores entre los que esta el analizador 2133.

En el apéndice D se explica parte de los comandos fundamentales de este programa.

4.4.1 Comunicación entre el robot y el WT9378

Hasta aquí se ha descrito cómo el programa WT9378 automatiza el proceso de obtención de los mapas de intensidad sonora, ahora es preciso detallar como se consigue que el WT9378 interactue con el equipo de medición y el robot.

Para que el programa WT9378 pueda controlar al robot, se proporciona dentro de él mismo, una "mascara de programa", cuyo código fuente se escribió en pascal; esta mascara permite insertar los programas que controlan al robot, que están escritos y compilados en lenguaje C de Microsoft.

Esta mascara de programa tiene el nombre de SETROBOT.PAS, los programas del robot son SETROB.C y ROBOT.C, el primer programa es el que se encarga de colocar la cabeza del robot y este a su vez a la sonda, en la posición (1,1), que es nuestra posición de origen. El programa ROBOT.C mueve la sonda a cualquier otra posición (x,y).

Para incrustar estos programas en el programa SETROBOT.PAS, se utilizaron los archivos SETROB.EXE y ROBOT.EXE, los cuales son llamados como procesos hijos del SETROBOT.PAS.

Para que el programa en Pascal llame a los programas ejecutables provenientes de rutinas en C, estas últimas son compiladas en modo de memoria sencillo, y son llamados mediante el procedimiento EXEC:

```
Exec(Path, CmdLine: string);
```

Cuando una ejecución del robot no es exitosa cualquiera de los dos programas del robot manda un número de error que se reporta como un 7 a través de la variable de entorno del MS-DOS conocida como DosError. Cuando el robot alcanza la posición se reporta en esta variable un valor de 0 Este valor es copiado a la variable interna de Turbo Pascal DosExitCode

Antes y después de usar EXEC debe utilizarse el procedimiento Swap Vectors, que se encarga de salvar o restaurar la tabla de vectores de interrupción del DOS, la cual es en caso de falla modificada por los procedimientos hijos. De no tomarse esta precaución, pudiera llegar a ocurrir la caída del sistema.

Este procedimiento es usado dos veces dentro del programa SETROBOT.PAS, primero para posicionar en (1,1) y después para moverse a una posición (x,y), definida.

A continuación se muestra el código de la mascara de programa SETROBOT.PAS que permite incrustar las rutinas hijas que controlan el robot⁴.

Programa SETROBOT.PAS

```
Program SetRobot;
```

```
var
  ProblemNo: Word; { =0 : No problems, >0 : problems. }
  Row      . Byte;
  Col      : Byte;
  TempDir  String[1];
  Direction Char;
```

```

PassNo : Byte; { 1: LF-pass, 2: HF-pass }
Code : Integer;
ExecCode : Integer;
TotalPoints : Word;
CurrentPoint : Word;
CommandString : String;
Line : array [1..9] of String;
RobotFile : Text;

```

```
begin
```

```
  ProblemNo:=0;
```

```
  { *** Get info from SetRobot.ini *** }
```

```
  Assign(RobotFile,'SetRobot.ini');
```

```
  Reset(RobotFile);
```

```
  Readln(RobotFile,Line[1]); { Total number text      }
```

```
  Readln(RobotFile,Line[2]); { Total number          }
```

```
  Readln(RobotFile,Line[3]); { Current point text   }
```

```
  Readln(RobotFile,Line[4]); { Current point number  }
```

```
  Readln(RobotFile,Line[5]); { Surface type. (Grid | Hemi | Any) }
```

```
  Readln(RobotFile,Line[6]); { Rows      (MaxSector)[1..12] }
```

```
  Readln(RobotFile,Line[7]); { Columns   (MaxPoint) [1..75] }
```

```
  Readln(RobotFile,Line[8]); { Dirs      [1..3]          }
```

```
  Readln(RobotFile,Line[9]); { Passes    [1..2]          }
```

```
  Close(RobotFile);
```

```
  Val(Line[4],CurrentPoint,Code);
```

```
  if CurrentPoint=0 then
```

```
    begin
```

```
      { *** Set up robot here *** }
```

```
      ProblemNo:=0;
```

```
      CurrentPoint:=1;
```

```
    end,
```

```
  if ProblemNo=0 then
```

```
    begin
```

```
      ProblemNo:=0;
```

```
      if ParamStr(1)='Reset' then
```

```
        begin
```

```
          { Final reset of robot here }
```

```
          { Llamada a programa SETROB.EXE }
```

```
          SwapVectors;
```

```
          Exec('SETROB41.EXE','');
```

```
          SwapVectors;
```

```
          ProblemNo := DosExitCode;
```

```
        end else if ParamStr(1)='F1' then
```

```
          begin
```

```
            { Set up to make F1 measurement }
```

```
            ProblemNo:=0;
```

```
          end else if ParamStr(1)='Calib' then
```

```
            begin
```

```
              { Set up to calibrate }
```

```

ProblemNo:=0.
end else
begin
  { *** Get Row, Col and Direction if possible *** }
  Val(ParamStr(1),Row,Code);
  if Code<>0 then
    ProblemNo:=251; { Tell WT9378 that the Row value is bad }

  if ProblemNo=0 then
  begin
    Val(ParamStr(2),Col,Code);
    if Code<>0 then
      ProblemNo:=252; { Tell WT9378 that the Col value is bad }
    end;

  if ParamCount>2 then
  begin
    TempDir:=ParamStr(3);
    Direction:=TempDir[1];
    if not (Direction in ['x','y','z']) then
      ProblemNo:=253; { Tell WT9378 that the Direction is bad }
    end else
      Direction:=' ';

  if ParamCount>3 then
    Val(ParamStr(4),PassNo,Code)
  else
    PassNo:=1;

  if ProblemNo=0 then
  begin
    { *** Move robot here *** }

    { Llamada a programa ROBOT.EXE }
    SwapVectors;
    { (x,y) = (ren,col) }
    Exec('ROBOT.EXE',ParamStr(2)+' '+ParamStr(1));
    SwapVectors;
    ProblemNo := DosExitCode;

    { ProblemNo:=0: ? }
    if ProblemNo=0 then
      CurrentPoint:=CurrentPoint+1;
    end,
  end;
end else
  if (ProblemNo>0) and (ProblemNo<254) then
  { ErrorMessage(Txt(ProblemNo)) };

if ProblemNo=0 then
begin
  Str(CurrentPoint,Linc[4]);
  Assign(RobotFile,'SetRobot ini');
  ReWrite(RobotFile);

```



```

Writeln(RobotFile,Line[1]); { Total number text }
Writeln(RobotFile,Line[2]); { Total number }
Writeln(RobotFile,Line[3]); { Current point text }
Writeln(RobotFile,Line[4]); { Current point number }
Writeln(RobotFile,Line[5]); { Surface type. (Grid | Hemi | Any) }
Writeln(RobotFile,Line[6]); { Rows      (MaxSector)      }
Writeln(RobotFile,Line[7]); { Columns  (MaxPoint)      }
Writeln(RobotFile,Line[8]); { Dirs     [1..3]          }
Writeln(RobotFile,Line[9]); { Passes   [1..2]          }
Close(RobotFile);
end;
Halt(ProblemNo); { Tell WT9378 about problems }
end.

```

Las partes en **negrita** son las líneas que se adicionaron al archivo fuente SETROBOT.PAS. Cuando este archivo se compilo se redujo el máximo de memoria disponible para asegurar, suficiente memoria para el proceso hijo, con la directiva de compilación {\$M \$4000,0,0}.

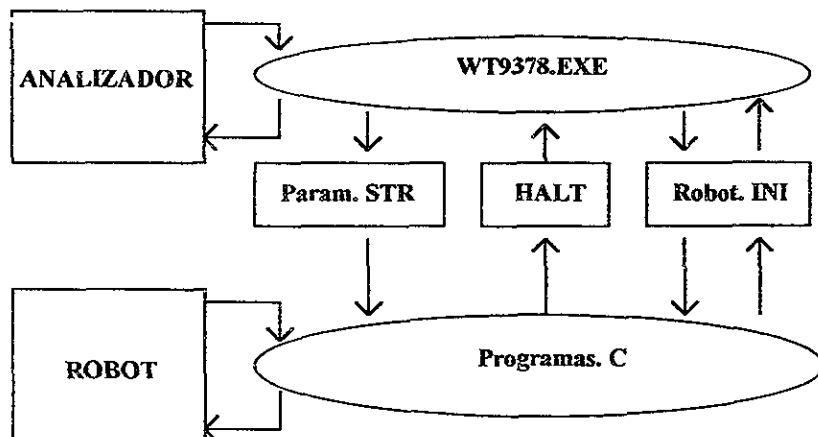
4.4.2 Integración y funcionamiento de los sistemas

El WT9378 esta conformado básicamente por los siguientes archivos:

- Un archivo ejecutable llamado WT9378.EXE, cuya descripción ya se menciona al principio de este subcapítulo.
- La mascara de programa SETROBOT.PAS.
- El archivo SETROBOT.INI ó InitFile, es el archivo de inicialización.
- El archivo ROBOT.INI, archivo de información.
- El archivo PARAM.STR, contiene información de posición, orientación, lugar actual.

Sabemos ya que el programa WT9378 EXE controla de manera automática al analizador y la sonda, además sabemos como se incrustan los programas de control del robot, ahora solo falta conocer como se lleva acabo el proceso de todo esto

El programa WT9378.EXE interactua con el programa SETROBOT.PAS, a través de los archivos de ROBOT.INI, PARAM.STR y de HALT (es el código de error), justamente como se ve en la siguiente gráfica.



El funcionamiento del sistema se produce al correr el programa WT9378.EXE, en donde se definen varios parámetros los cuales ya se discutieron, inmediatamente el programa llamará a SETROBOT.EXE, mediante el archivo SETROBOT.INI, este último archivo permite enviar información sobre el total de puntos que serán analizados, el tipo de superficie (en nuestro caso es una rejilla), el número de renglones, el número de columnas y la dirección del barrido. El archivo PARAM.STR manda la posición actual del robot. Mediante la línea de comandos de SETROBOT se envía la orden de posicionamiento inicial (1,1) o bien a la posición a la que deberá moverse el robot (renglón, columna, dirección). Ahora bien cuando se ha arrancado el sistema este mandará colocar a la sonda en la posición (1,1), tomará el dato y después mandará la nueva orden correspondiente a la nueva posición del robot, cuando se termine el barrido de todos los puntos de medición (superficie definida previamente), se puede mandar almacenar esta información o bien procesarla para poder visualizar inmediatamente nuestro mapa de intensidad sonora, o bien guardar la información como un archivo de texto o de extensión. TXT, para después poder procesarla en el programa para MATLAB y conseguir con esto un análisis diferente de nuestros datos.

Lo más saludable y recomendable es consultar el manual de usuario del programa WT9378, si es que se desea profundizar a un más; en el apéndice D se describe un poco más.

REFERENCIAS CAPITULO 4

1. J. M. Poblet, *Interconexión de periféricos a microprocesadores*, (Ed Marcombo), (1984)
2. Brüel & Kjær, *Instruction Manual Analyzer type 2133*, Technical Documentation.
3. Brüel & Kjær, *Sound Intensity*, Technical Documentation.
4. Brüel & Kjær, *Mapping and Sound Power Program WT9378*, Technical Documentation, (1993).

CAPITULO 5

5. PRUEBAS DEL SISTEMA

Para probar el sistema se requieren los siguientes elementos:

- 1.- Una computadora personal (PC), y el programa WT9378
- 2.- Una tarjeta PCL-720.
- 3 - Dos tarjetas PCLD-782
- 4.- Una tarjeta NI-488 2, junto con su programa de manejo
- 5.- Un circuito de control, del robot, con sus programas de control.
- 6.- Un sistema mecánico (robot).
- 7 - Un analizador de frecuencias, tipo 2133.
- 8.- Una sonda de intensidad sonora tipo 3545 y la unidad de control remoto ZH0354
- 9.- Cables de conexión

Es necesario conectar todo el equipo, identificando para ello las terminales correspondientes, como se indica a continuación.

a) conectar la llave (electrónica) de acceso al programa WT9378, en el puerto paralelo.

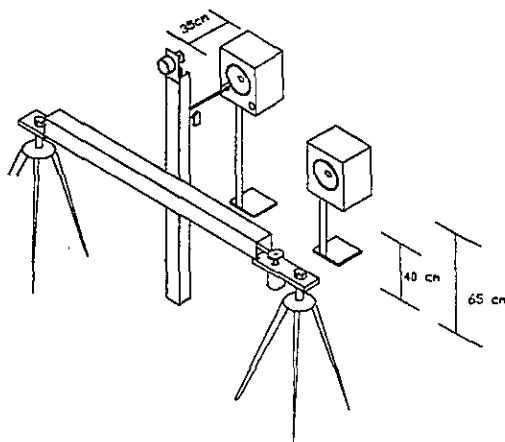
b) conexiones del sistema

- 1.- sobre la llave conectada al puerto paralelo, hay que conectar el cable marcado como CPPA en el conector DB25 en la tarjeta de control llama MOTORXY MOTORXY.
- 2 - identificar las terminales de la tarjeta PCL-720 (en el gabinete de la PC), marcadas como A, B y C. La terminal A, se conecta al circuito EJEX, la terminal B, se conecta al circuito EJEY, la terminal C, se conecta a la terminal CLK, del circuito MOTORXY.
- 3.- del robot salen tres cables llamados MOTX, MOTY y POSI. que van al gabinete que contiene los circuitos de control del motor, en este lugar están tres terminales, dos son de conectores para dos pines y una de cuatro pines, la terminal de cuatro pines se conecta al cable POSI, el cable MOTX se conecta a la terminal MX, y el MOTY a la terminal MY
- 4.- en la PC, esta la terminal de la GPIB, la cual se conecta a la terminal del analizador mediante el cable marcado como CANALI.
- 5.- conectar al control remoto de la sonda la terminal indicada en el analizador.
- 6 - acoplar el control remoto y la sonda en el "carro porta sonda" del robot, mediante los tornillos respectivos
- 7 - Conectar la sonda al control remoto
- 8.- Conectar el equipo a la toma de corriente de 120 Volts, 60 Hz.

- c) cargar el programa WT9378.
- d) efectuar mediciones, y análisis de resultados.

Una vez hechas las conexiones necesarias y para efectuar las primeras mediciones es necesario configurar el programa WT9378, bajo las siguientes condiciones: definir el modo de robot, especificar que el barrido es sobre una rejilla, especificar el tamaño de la rejilla, introducir el valor de la presión atmosférica, la temperatura, el tamaño del separador en la sonda y finalmente definir en que eje se tomaran las lecturas de intensidad. Es recomendable consultar el apéndice D y el manual de instrucción del programa WT9378.

La primera prueba de este proyecto fue hacer en forma repetida mediciones sobre dos fuentes (baffles excitados con una señal de ruido rosa), se hizo un barrido de 16x12 (renglón, columna). El esquema de la prueba fue el siguiente.



El propósito fue comprobar el correcto funcionamiento del sistema, obteniéndose resultados aceptables, los mapas realizados con esta prueba, son los incluidos en el capítulo 1 (mapa 1, 2 y 3).

Los resultados obtenidos:

- Se determino que el sistema, podrá entregar en aproximadamente 35 min, mapas de 16x12 en los diferentes formatos de presentación
- El equipo respondió correctamente cuando se provocaron errores, como lo son, el que el sistema mecánico no pueda colocar la sonda en el punto correcto, el sistema de adquisición no recibe ni manda datos, el puerto paralelo manda datos aleatoriamente.

5.1 CALIBRACION¹

Para que la medición de intensidad sonora tuviera aceptabilidad, fue necesario calibrar al sistema de medición, para esto se utilizó el calibrador de intensidad sonora tipo 3541 de B&k, este equipo permitió calibrar los niveles de presión sonora, los de intensidad sonora y de velocidad de partícula, además del índice residual de presión-intensidad y del índice residual de intensidad. Para ello se empleó lo siguiente:

- Pistófono, tipo 4228 B&K. Se emplea para la calibración de presión y de intensidad. El pistófono genera una onda de presión con nivel de 118 dB re 20 μ Pa y 250 Hz, con estos parámetros se ajustaron las sensibilidades de cada canal del analizador de frecuencias.
- Acoplador, tipo UA0914 B&K. Consiste de dos cámaras conectadas por un elemento de acoplo. Cuando el pistófono es conectado al acoplador, se generan presiones sonoras de diferente fase entre la cámara superior y la inferior, ver figura 5.2.1. Dentro del acoplador se simula una onda sonora plana, que se propaga en un campo libre. Si los micrófonos son colocados dentro de la misma cámara entonces es posible calibrar la sensibilidad de la presión, por que los dos micrófonos registraran la misma presión y la misma fase.

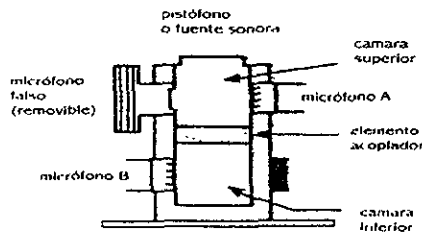


Figura 5.2.1 Acoplador, se muestran los componentes que se conectan a este elemento.

- Fuente sonora de banda ancha, tipo Z1 0055 B&K. Esta se utiliza para medir el índice residual presión-intensidad, el cual es usado para estimar la aceptación de la medición.
- Carta de calibración. Proporciona los niveles que se tienen que registrar en la calibración, si estos difieren, entonces se tendrá que hacer los respectivos ajustes de acuerdo con la información contenida en esta carta.
- Barómetro de corrección. Se utiliza para hacer las correcciones debidas a los cambios de presión atmosférica.

El índice residual de presión-intensidad de un sistema de medición se define como la diferencia entre los niveles de intensidad y de presión indicados cuando se les aplica la misma presión a los micrófonos en un mismo ángulo de incidencia. Idealmente cuando a los micrófonos se les aplica una misma señal de presión a un ángulo de 90° sobre el eje de la sonda el nivel de intensidad es cero. Pero en realidad los micrófonos tienen pequeñas diferencias entre sus respuestas de fase lo que causa una diferencia de fase entre las señales de los micrófonos, por lo que el equipo de medición indicara un

nivel de intensidad diferente de cero. Entonces el índice residual de presión-intensidad es:

$$L_{k,0} = L_{p,0} - L_{I,0} \quad (5.1)$$

donde:

- $L_{k,0}$ es el nivel del índice residual presión-intensidad.
- $L_{p,0}$ es el nivel de presión sonora.
- $L_{I,0}$ es el nivel de intensidad sonora.

Este dato sirve para evaluar la aceptación de una medición, corregir y optimizar la aceptación del sistema de medición y para corregir los errores debidos a estas desviaciones de fase. Para lo anterior se tiene que calcular el índice residual del nivel de intensidad usando la siguiente ecuación:

$$L_{I,R} = L_p - L_{k,0} \quad (5.2)$$

donde:

- $L_{I,R}$ es el índice residual del nivel de intensidad en una medición.
- L_p es el nivel de presión sonora en la misma medición.
- $L_{k,0}$ es el valor del índice residual de presión-intensidad para el equipo obtenido previamente.

Para asegurar que una medición tiene un error menor a 1 dB, el nivel de intensidad sonora en la medición tendrá que ser 7 dB por arriba del nivel de intensidad residual:

$$L_I - L_{I,R} > 7\text{dB para un error} < 1\text{dB.}$$

Para la corrección del nivel de intensidad debida al error de fase se puede emplear la siguiente formula:

$$I_{cor} = I_m - \frac{p_m^2 I_0}{p_0^2} \quad (5.3)$$

donde:

- I_{cor} es la intensidad corregida.
- I_m intensidad en la medición.
- p_m presión en la medición.
- I_0, p_0 intensidad y presión para igual presión aplicada en ambos micrófonos

La temperatura, la presión atmosférica (factores que afectan ala densidad del aire) y el espaciador o separador de los micrófonos, también afectan las mediciones de intensidad sonora por lo que el equipo de medición se tiene que ajustar. Las condiciones nominales de calibración son las siguientes:

Presión: 1013 hPa

Temperatura: 20°C

Humedad: 65% RH

Si las condiciones para la medición difieren de las nominales entonces se tienen que hacer los siguientes ajustes:

Si el espaciador no es el de 50 mm:

- Se tiene que aplicar los siguientes niveles a la corrección de acuerdo a lo que se este midiendo, como se indica en la tabla.

Espaciador	Presión	Velocidad	Intensidad
12 mm	0 dB	+ 12.4 dB	+ 6.2 dB
6 mm	0 dB	+ 18.4 dB	+ 9.2 dB
Otros	0 dB	Ver ecuación 5.4	Ver ecuación 5.5

$$20 \log \frac{50}{\Delta r_0 (\text{mm})} \text{ dB} \quad (5.4)$$

$$10 \log \frac{50}{\Delta r_0 (\text{mm})} \text{ dB} \quad (5.5)$$

donde

Δr_0 es la longitud de espaciador o separador que se empleara en la medición.

Si la presión ambiente no es de 1013hPa:

- El nivel de corrección para presión sonora se muestra en el barómetro UZ 0004.
- El nivel de corrección para velocidad de partícula es numéricamente igual el nivel de corrección de presión sonora pero de signo opuesto.
- La corrección para el nivel de intensidad sonora es 0 dB.

Si la temperatura T, no son 20° C:

- El nivel de corrección para presión sonora es 0 dB.
- El nivel de corrección para velocidad de partícula esta determinado con la siguiente ecuación:

$$0.05 \frac{\text{dB}}{^\circ\text{C}} * (T - 20)^\circ\text{C}$$

- El nivel de corrección para la intensidad sonora se determina con la siguiente ecuación:

$$0.024 \frac{\text{dB}}{^\circ\text{C}} * (T - 20)^\circ\text{C}$$

Si la humedad no es del 65% RH: No se requieren correcciones.

Para mayores detalles de la calibración es recomendable consultar los manuales del calibrador tipo 3541 y del analizador 2133

5.2 PRUEBA DE CARACTERIZACION DE LA GUITARRA ACUSTICA CLASICA

Objetivo: Es el de probar el funcionamiento del sistema, al someterlo a condiciones y factores que influyen en una medición real, como fue el intentar caracterizar una guitarra acústica clásica, por medio de los mapas de intensidad obtenidos.

Se ha determinado experimentalmente que el vientre de la guitarra acústica clásica vibra a frecuencias menores a 500-600 Hz en sólo cinco modos de vibración², aproximadamente a 200, 300, 400, 500 y 600 Hz. La prueba consistió en intentar obtener los mapas de intensidad para la guitarra, en algunos de sus modos de vibración.

La fuente fue colocada dentro de una cámara anecoica, por que en un recinto como este no existen ondas reflejadas, por lo tanto se puede pensar que la única energía medida será la radiada por el cuerpo de la guitarra.

Las condiciones de la medición son las siguientes:

$$1hPa = 1 \text{ mbar}$$

Presión: 769 hPa o 769 mbar

Temperatura 23°C

Humedad: desconocida.

Espaciador: 50mm

Bajo estas condiciones fue necesario calibrar el equipo:

Para calibrar presión:

El nivel nominal del calibrador es: 118.0 dB re 20 μ Pa

Corrección del barómetro: - 2.4 dB

Corrección por temperatura: 0 dB

corrección por el separador: 0 dB

El equipo tiene que indicar: 115.6 dB

Obviamente al analizador se le introducen los parámetros ambientales y este a su vez corrige el valor de la densidad. Si el analizador no indica el nivel de presión esperado, en este caso 115.6 dB, se tienen que ajustar los niveles de sensibilidad de cada canal del analizador, por medio de la auto calibración o de forma manual hasta que los canales queden debidamente ajustados al valor esperado.

Para calibrar el nivel de intensidad:

El nivel nominal del calibrador es: 117.9 dB re 1 pW/m^2

Corrección del barómetro: 0 dB

Corrección por temperatura: $(23-20) \times (0.024) \text{ dB} = +0.072 \text{ dB}$

Corrección por el separador: 0 dB

El equipo tiene que indicar: 117.97 dB

Como la calibración de presión fue adecuada y los parámetros ambientales introducidos al analizador, el equipo indico el valor esperado sin ningún problema, en este caso 117.97 dB.

Para calibrar velocidad de partícula:

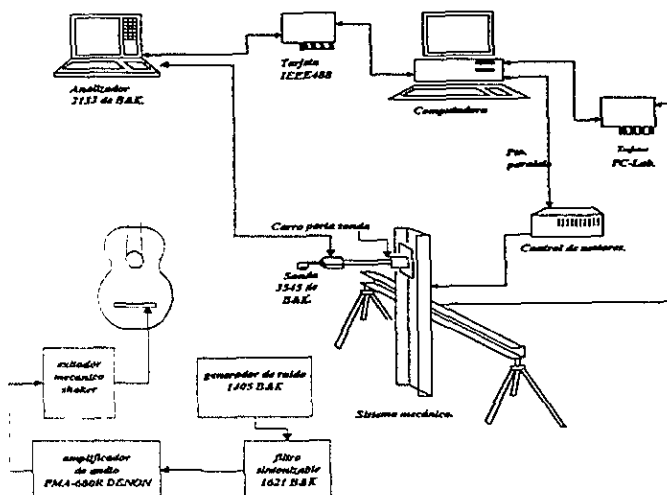
Este paso no se efectuó por que queda claro que si el nivel de intensidad fue el esperado no era necesario calibrar al equipo para esta medida.

El analizador de frecuencias 2133, fue programado para efectuar una promediación de tipo lineal de 2s, por cada punto de medición sobre la rejilla.

El equipo adicional (al descrito al principio del capítulo) utilizado, para la prueba fue:

- Amplificador de medición 2623 de B&K.
- Filtro paso banda sintonizable 1621 de B&K.
- Generador de ruido 1405 de B&K.
- Amplificador DENON PMA-680R
- Excitador mecánico 4810 de B&K.
- Micrófono de condensador 4163 de B&K.
- Guitarra clásica marca: Yamato.

Enseguida se muestra el diagrama a bloques de lo que fue el experimento.



La guitarra fue montada en una base diseñada para este propósito. En la figura 5.2.1 se muestra el diseño y en la figura 5.2.3 se muestra la fotografía con la guitarra ya montada.

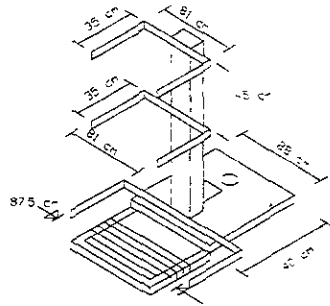


Figura 5.2.1 Base sujetadora de la guitarra.

Las lecturas de la energía radiada por la guitarra fueron captadas por la sonda a una distancia de 35 cm, entre ambos elementos como muestra la siguiente fotografía.

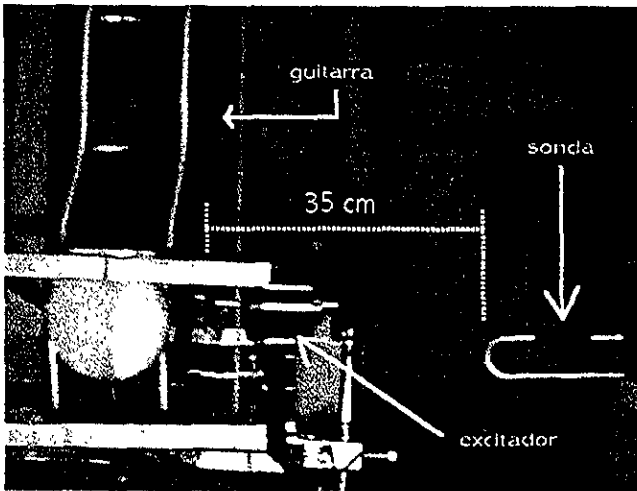


Figura 5.2.2 se muestra la separación entre la guitarra y la sonda al momento de efectuarse la prueba

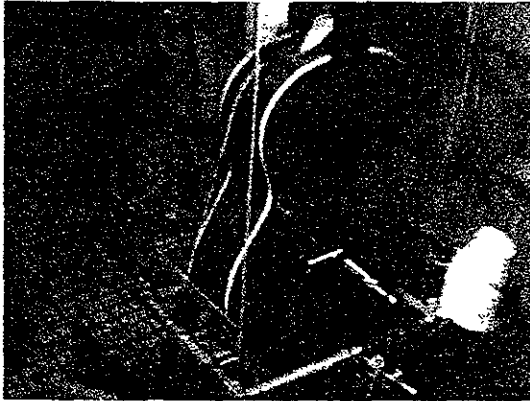


Figura 5.2.3 Se muestra la base para sujetar la guitarra, también se nota al excitador mecánico.

El sistema mecánico tiene el siguiente aspecto físico (figura 5.2.4).

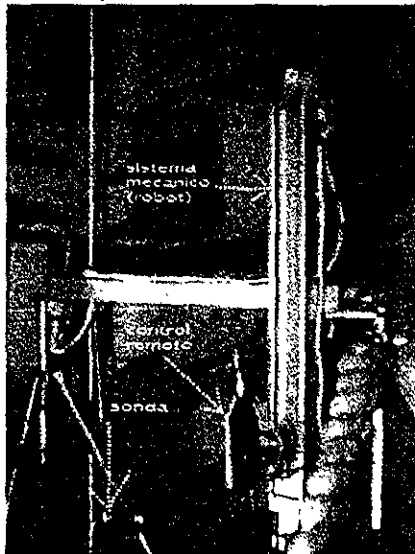


Figura 5.2.4 se muestra el robot, la sonda y su unidad de control remoto.

La guitarra fue forzada a radiar sonido, mediante el generador de ruido 1405, se selecciono el modo de ruido rosa de ancho de banda de 50 kHz. Esta señal de ruido se hizo pasar por el filtro sintonizable 1621 de B&K, las frecuencias centrales fueron seleccionadas

de acuerdo a las frecuencias en donde se suponen están los modos de vibración. Con la ayuda del amplificador de medición 2636, un micrófono puesto dentro de la cámara y un audífono se localizaron las frecuencias en que existía mayor presión sonora y una mayor captación acústica por parte del oído. Estas frecuencias fueron las frecuencias centrales del filtro y están comprendidas dentro de las frecuencias que indican los investigadores¹ para los modos de vibración que se obtuvieron.

Una vez filtrada la señal, se paso por el amplificador de audio PMA-680R (amplificador de audio domestico). La señal ya amplificada por un factor de 15X (este amplificador consigue hasta 40X) se conecto al excitador mecánico, el cual, produce vibraciones en correspondencia con la señal de entrada. Esta vibración era transmitida al cuerpo de la guitarra, exactamente en el puente entre la 2ª y 3ª cuerda, (el ruido que se escucho es muy semejante al ruido que emite un calentador de gas).

La figura 5.2.5 nos muestra el área de la guitarra que fue barrida con la sonda y de esta manera obtenidos los mapas que más adelante se muestran. También se trata de ilustrar el lugar en el que el excitador mecánico fue colocado, par forzar a que emitiera sonido la guitarra



Figura 5.2.5 Se muestra el área de la guitarra, que fue mapeada, la parte, circular luminosa es el excitador mecánico.

Después de instalar y checar los equipos se procedió a configurar el sistema de automatización, para comenzar la medición.

Los resultados de estas pruebas son los siguientes

Con la señal filtrada a 160 Hz, se obtiene el mapa a 163 Hz (por que se utiliza una resolución de 1/12 de octava):

Promediación: lineal Factor de amplificación de la señal de exitación: 15x
Intensidad: 120.5 dB+

ESTA TESIS NO DEBE
SALIR DE LA BIBLIOTECA

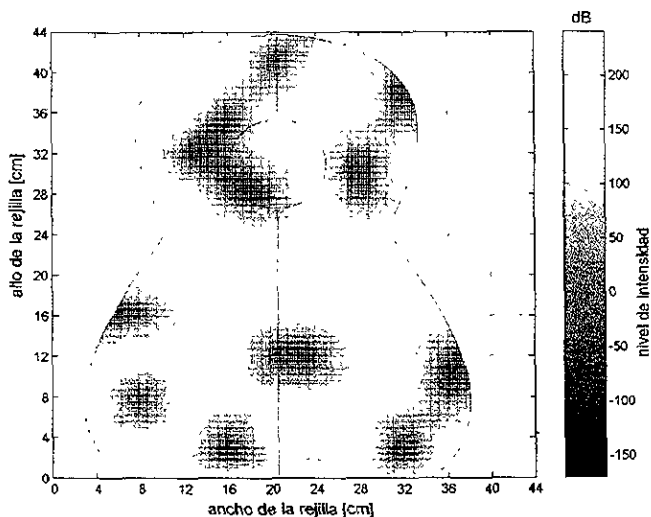


Figura 5.2.6 Mapa de intensidad sonora para la frecuencia de resonancia a 163 Hz. Las partes claras indican radiación de energía, las oscuras indican zonas de absorción de energía.

De acuerdo a un estudio realizado por Molin y Stetson³, que consistió en soportar muy rígidamente la tapa superior de la guitarra, por su borde y forzándola a vibrar en diferentes frecuencias, excitándola en distintos puntos mediante un vibrador electro-mecánico. Utilizando interferometría holográfica (método óptico para detectar desplazamientos), fueron registrados los modos normales de vibración, como el que se muestra en la figura 5.2.7.

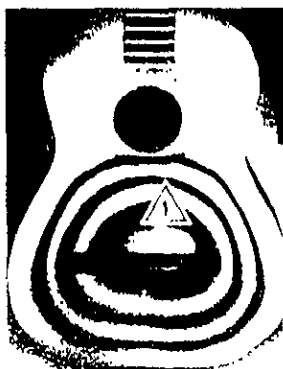
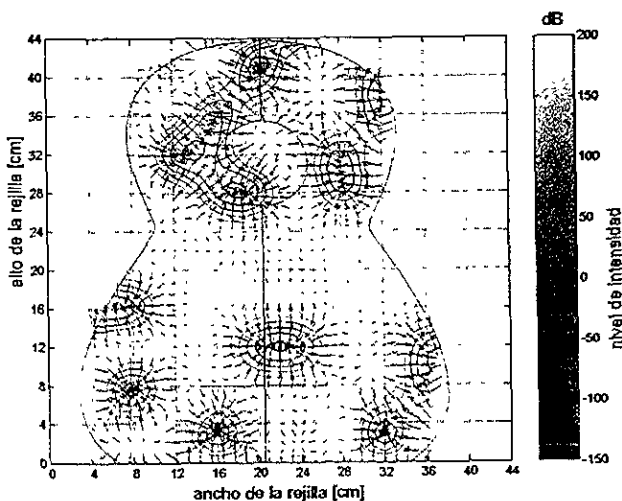


Figura 5.2.7 interferograma holográfica de la frecuencia de resonancia a 185 Hz encontrado por Molin y Stetson (el triángulo indica el punto de excitación), las líneas negras indican zonas de igual amplitud.

Si comparamos las dos ultimas figuras notamos algunas similitudes. La primer frecuencia de resonancia encontrada por interferometría esta a 185 Hz, un patrón de radiación muy similar al de la figura 5.2.7 encontrado por el método de intensidad fue a 163 Hz (figura 5.2.6), obviamente no son iguales por que los modos de vibración están determinados por la estructura y propiedades de la madera (se desconoce el fabricante de la guitarra empleada en los experimentos de Molin y Stetson, y los detalles de las mismas), aun que se nota alguna similitud, justamente en la zona del puente.

El método de la intensidad sonora entrega el nivel de intensidad, indicándonos el flujo neto de energía acústica que esta emitiendo la fuente sonora. El método de interferometría utilizado, únicamente nos indica la forma de vibración de la tapa de la guitarra, y no entrega ningún nivel, por lo que se desconoce que tanta energía se esta liberando de la fuente.

Otra presentación de los mapas es en forma de vectores, por lo que en la siguiente gráfica se muestra el mapa vectorial y de contorno para esta prueba.



La segunda prueba fue el encontrar el segundo modo de vibración, teniendo como referencia la frecuencia de resonancia encontrada por Molin y Stetson a 287 Hz. El resultado es el siguiente:

Con la señal filtrada a 220 Hz, se obtiene el mapa a 200 Hz:

Promediación: lineal.

Factor de amplificación de la señal de excitación: 15x

Intensidad: 60.2 dB+

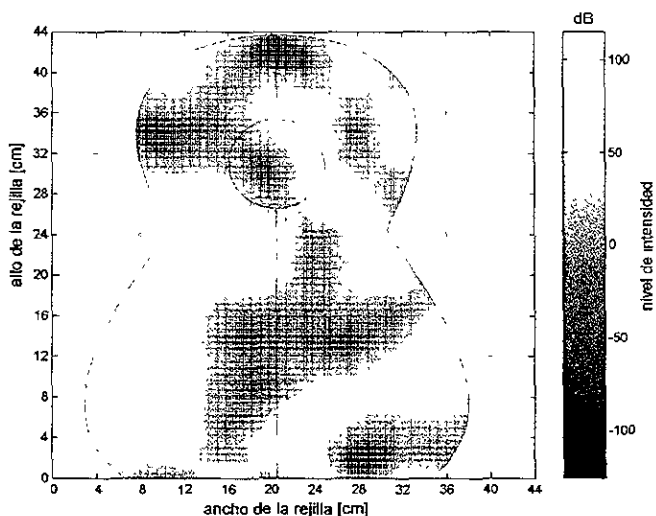


Figura 5.2.8 Mapa de intensidad sonora para la frecuencia 200 Hz

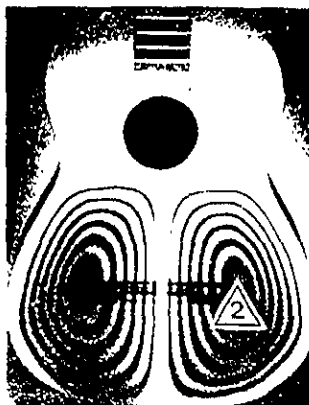
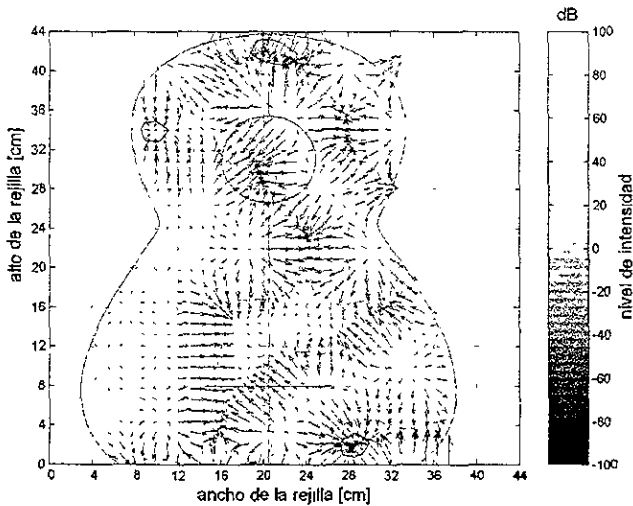


Figura 5.2.9 interferograma holográfica de la frecuencia de resonancia a 287 Hz (el triángulo indica el punto de excitación), las líneas negras indican zonas de igual amplitud

En la figura 5.2.9 se notan dos zonas de radiación de energía, el mapa de intensidad obtenido (figura 5.2.8), marca también dos zonas, una de radiación (zona amarilla) y otra de absorción de energía (zona roja), las frecuencias que se obtiene, en ambos métodos no son iguales, pero si cercanas, se tomaron varias medidas en frecuencias cercanas a lo que marcan Molin y Stetson (287 Hz), pero no se obtuvieron mapas que marcaran dos zonas de radiación, a excepción del de la figura 5.3.8 a 200 Hz.

El mapa vectorial y de contorno se muestra enseguida:

El mapa vectorial se muestra enseguida.



El tercer modo de vibración según Molin y Stetson, esta en 460 Hz. sobre la base de esto se encontró lo siguiente:

Con la señal filtrada a 500 Hz. se obtiene el mapa a 487 Hz:

Promediación: lineal. Factor de amplificación de la señal de excitación: 15x
Intensidad: 64.0 dB+

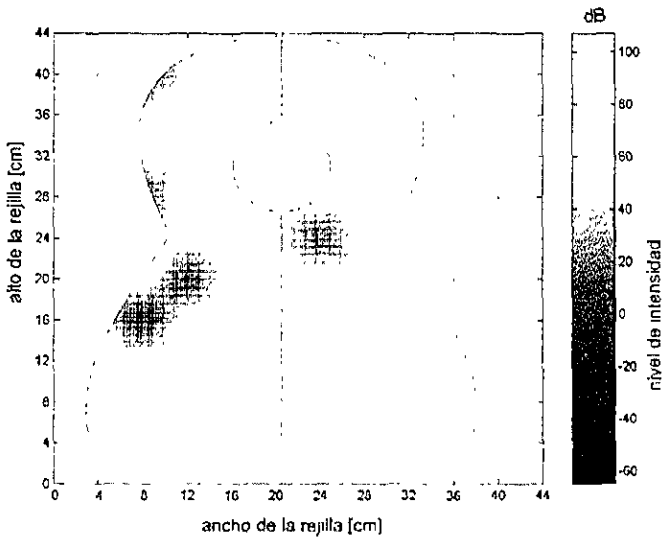


Figura 5.2.10 Mapa de intensidad sonora para la frecuencia 487 Hz

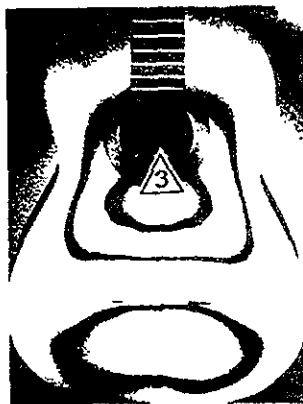
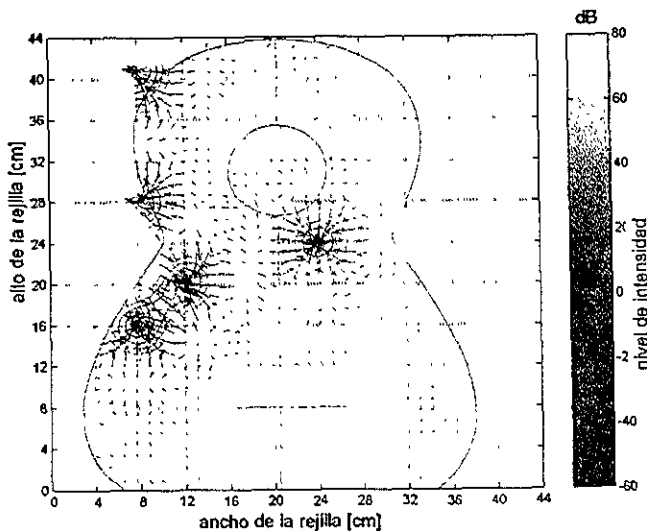


Figura 5.2.11 interferograma holográfica de la frecuencia de resonancia a 460 Hz (el triángulo indica el punto de excitación), las líneas negras indican zonas de igual amplitud. El mapa vectorial y de contorno de esta prueba es el siguiente:



En este caso los mapas no guardan mucha similitud, pero los mapas cercanos a los 487 Hz son completamente diferentes, por lo que se eligió presentar los mapas anteriores.

Los resultados anteriores son únicamente una muestra de lo que el sistema puede realizar. Se acepta, que los resultados encontrados mediante este método no concuerdan

con los de interferometría, pero si cumplen satisfactoriamente con el propósito de la tesis, al proporcionar un medio totalmente automático para la elaboración de mapas de intensidad sonora.

Los resultados pueden ser por mucho mejorados, considerando que se dejan de lado el índice de presión-intensidad, el índice residual del nivel de intensidad, las correcciones debidas a la fase, probar con otra distancia entre la guitarra y la sonda, medir con otro separador, cambiar el punto de excitación a la guitarra y aumentar el tiempo de promediación.

REFERENCIAS CAPITULO 5

1. Brüel & Kjær, Sound intensity calibrator type 3541, Technical Documentation.
2. A. Caldersmith, The guitar frequency response, Journal of Guitar Acoustic, Sept. (1982)
3. E.V. Jansson, A study of acoustical and hologram interferometric measurements of the top plate vibrations of a guitar, Acustica. Vol. 25 (1971)
4. Brüel & Kjær, Mapping and Sound Power Program WT9378, Technical Documentation, (1993).

CAPITULO 6

6. CONCLUSIONES

En seguida se indica varias conclusiones obtenidos en este trabajo de tesis:

- Se cumplió con el objetivo, de construir un sistema automático (robot) para la colocación de una sonda de intensidad en el plano $X-Y$, por lo que se pudo automatizar el proceso de mapeo de intensidad sonora. El sistema fue construido con materiales de bajo costo y fácil disponibilidad en el mercado nacional.
- El robot tiene la ventaja, sobre el posicionamiento manual de la sonda de intensidad sonora, la comodidad en su uso y por lograr una mejor repetibilidad de resultados.
- El robot, permite ser utilizado para otros propósitos, ya que los programas de este como los de todo el sistema son independientes, lo que permite su utilización para cualquier otro fin.
- Los programas de control pueden ser modificados ó ampliados para controlar cualquier otro dispositivo o actuador y no necesariamente motores de cd.
- Se consigue elaborar mapas con relativa facilidad para cualquier fuente sonora, tomando las consideraciones pertinentes para las mismas.
- La instalación de equipo es muy sencilla, la transportación también, a excepción de la transportación del analizador de espectros, que implica esfuerzo y mucho cuidado por ser un equipo de instrumentación delicado y demasiado caro.
- Como defectos se tiene que, cuando el robot envía un mensaje de error al programa WT9378, se detiene todo el proceso de mapeo, acción que causa una caída del sistema por lo que es necesario reinicializar la computadora, lo único posible para acabar con este tipo de errores es hacer más eficiente el funcionamiento del equipo, y modificar los programas de control y los de adquisición de datos, aun que hay cosas que no se pueden modificar.
- La detección de las posiciones llega a tener problemas cuando el voltaje de la línea es bajo, por que se trata de proporcionar una corriente constante al circuito respectivo, la posible solución, es el empleo de algún microcontrolador que mande en forma serial la información de los sensores al puerto serie de la computadora.

- Como observación personal, quedo satisfecho con el trabajo, sabiendo que es posible mejorar cualquiera de las partes del proyecto (programas, circuitos electrónicos, partes mecánicas y mediciones). Dado que el laboratorio carecia de un equipo de este tipo principalmente por el costo, en lo personal espero que sea de utilidad.

APENDICE A

Listados de los programas de control del robot Hechos en lenguaje C ver. 7 de Microsoft.

I.- Programa SETROB.C

```

/* Programa llamado SETROB. C el cual coloca a la sonda en la posición (1,1)*/

#include<stdio.h>
#include<dos.h>
#include<conio.h>
#include<time.h>
#include<stdlib.h>

void delay(clock_t wait); /* prototipo en delay */
Lugar_cero(int a,int c); /* prototipo en Lugar_cero */

int monitor; /* variable global */

/*****PROGRAMA PRINCIPAL*****/
main()
{
    int a,b,c,d;
    monitor=1;

    /*+++++ Programacion del timer 8253 como monoestable a 10 segundos +++++*/
    /* los 10 segundos son suficientes para que la sonda sea colocada en la posición (1,1)*/
    /* estando esta en cualquier otra posición, aun siendo esta la más alejada del origen */

    outp(0x2A0+7,0x32); /* Byte de control,contador 0,carga LSB y MSB*/
                        /* en modo 1, BCD 0 */
    outp(0x2A0+4,0xF4); /* carga LSB */
    outp(0x2A0+4,0x7E); /* carga MSB, 25000 pulsos * 0. 4 mseg = 10 seg */

    outp(890,11); /* genera un flanco positivo en el bus de control */
    outp(890,10); /* de puerto paralelo para disparar el timer*/

    delay((clock_t)4); /* retraso para que se pueda dar el cambio de nivel*/
                        /* del timer, solo para maquinas 486, 2 millseg. */

    /*+++++ Inicio del programa +++++*/

    a=inp(0x2A0+0); /* Lectura de la tarjeta PCL-720 en el segmento a*/
                  /* de la tarjeta 1 para localizar la posición inicial */
    c=inp(0x2A0+2); /* Lectura de la tarjeta 2 PCL-720 en el segmento c*/
                  /* para localizar la posición inicial del eje Y*/
    Lugar_cero(a,c); /* llamada a la función que busca el origen */

    if(monitor==2) /* monitor es una variable, cuyo valor proviene de la función anterior*/
                  /* servira para determinar si en el proceso de posicionamiento ocurrió un error*/
    {
        printf("ERROR ERROR");
        outp(888,0);
        return (7); /* regresa un código 7 al sistema, este código lo lee el programa WT9378*/
                   /* y lo interpreta como un error, de manera que se detiene el proceso de mapeo*/
    }
    else
    {
        outp(888,0);
        return (0);
    }
}

/*+++++ Rutina que localiza la posición inicial+++++*/

```

```

Lugar_cero(int a,int c)
{
  int mon;

  outp(888,0); /* coloca en el bus de datos a todos sus bits en 0 cuya acción es la*/
               /* no activación de los motores*/
/* Lectura de los segmentos a del eje X y c del eje Y, por que dentro de estos se encuentra la posición*/
/* inicial*/
  a= inp(0x2A0+0);
  c= inp(0x2A0+2);

  mon=inp(889); /* mon es la variable que contiene el nivel del monoestable*/

  while (a!=0x01 | c!=0x01) /* mientras los dos segmentos no indiquen (1,1) hace lo siguiente*/
  {
    if(mon == 0x87) /* si el monoestable tiene un nivel bajo efectua lo siguiente*/
    {
      delay((clock_t)1);
      a=inp(0x2A0+0);
      c=inp(0x2A0+2);
      if(c != 0x01) /* pregunta si en Y se alcanza la posición inicial */
      {
        outp(888,4); /* activa el motor 2 atras*/
        if(a != 0x01) /* pregunta si en X se alcanza la posición inicial */
        {
          outp(888,5); /* activa los motores 1 y 2 atras*/
          a=inp(0x2A0+0);
          c=inp(0x2A0+2);
        }
        else
        {
          outp(888,4); /* si en X es la posición 1 mantiene activado el motor 2*/
          a=inp(0x2A0+0);
          c=inp(0x2A0+2);
        }
      }
      else
      {
        /* si en Y es la posición inicial 1 se efectua lo siguiente*/
        if(a != 0x01) /* pregunta si en X se alcanza la posición inicial*/
        {
          outp(888,1); /* como en Y, es 1, se activa unicamente el motor 1 atras*/
          a=inp(0x2A0+0);
          c=inp(0x2A0+2);
        }
        else
        {
          outp(888,0); /* si en X es 1, y como Y ya es 1 se logra que la posición sea (1,1)
                       /* deteniendo los motores*/
          a=inp(0x2A0+0);
          c=inp(0x2A0+2);
        }
      }
      mon=inp(889); /* al termino del ciclo se vuelve a leer mon*/
      a=inp(0x2A0+0);
      c=inp(0x2A0+2);
    }
    else /* si resultado que mon tiene un nivel alto se hace lo siguiente*/
    {
      outp(888,0); /* detiene motores*/
      monitor=2; /* coloca el valor de 2 a la variable monitor*/
      break; /* y rompe el while*/
    }
  } /* Fin while */
}

/*+++++++Rutina que proporciona retrasos en milisegundos+++++++*/
void delay(clock_t wald)
{
  clock_t goal;

```

```

goal = wait + clock();
while (goal > clock() )
:
}

```

II.- El programa ROBOT.C

La presentación de este listado es una presentación reducida del programa original. Únicamente comprende el movimiento en el cuadrante 3 (ver figura 3.3.9), para una sola acción motor X atrás, motor Y abajo (ver tabla 3.1)

```

/* Programa llamado ROBOT.C sirve para posicionar a la sonda en cualquier punto del plano */

#include <process.h>
#include <stdlib.h>
#include <time.h>
#include <dos.h>

/* Declaración de subrutinas */

void delay(clock_t wait);
void lugar_cero(int a,int c);
void destino(int x,int y);
void cuadrante_1(int a,int c,int x,int y,int cuadx,int cuady);
void cuadrante_2(int a,int d,int x,int y,int cuadx,int cuady);
void cuadrante_3(int b,int d,int x,int y,int cuadx,int cuady);
void cuadrante_4(int b,int c,int x,int y,int cuadx,int cuady);
void accion_1(int x,int y,int cuadx,int cuady);
void accion_2(int x,int y,int cuadx,int cuady);
void accion_3(int x,int y,int cuadx,int cuady);
void accion_4(int x,int y,int cuadx,int cuady);
void robot_scan(int x,int y);

/* arreglos estaticos, con las posiciones que toma el robot, en el eje X
y en el eje Y */

static int equis[17] = { 0x0,0x01,0x02,0x04,0x08,0x10,0x20,0x40,0x80
,0x01,0x02,0x04,0x08,0x10,0x20,0x40,0x80};
static int ye[17] = { 0x0,0x01,0x02,0x04,0x08,0x10,0x20,0x40,0x80
,0x01,0x02,0x04,0x08,0x10,0x20,0x40,0x80};

int cuadx,cuady; /* variable global */
int monitor;

/*****PROGRAMA PRINCIPAL*****/
main (int argc, char *argv[])
{
int t;
int x,y;

monitor=1;
if (argc<3) return (1); /* parametros insuficientes */
if (argc>3) return (2); /* demasiados parametros */
x = atoi (argv[1]);
y = atoi (argv[2]);

robot_scan(x,y);

if(monitor==2)
{
printf("\n\nOcurrio un ERROR \n");
printf("\n para corregir e iniciar de nuevo:\n");
printf("\n a.- Reinicializa a la PC.\n");
printf("\n b.- Checa la parte mecanica y corrige.\n");
delay((clock_t)1000);
outp(888,0);
}
}

```



```

        return (7);
    }
else
    {
        delay((clock_t)850);
        outp(888,0);
        /*printf("\t\tPOSICION INICIAL ENCONTRADA\n");*/
        return (0);
    }
}

/* procedimiento en el cual se efectua la programación del timer y se reconoce la posición actual*/
/* ó posición activa de la sonda, las variables x, y son las coordenadas de la posición deseada*/
void robot_scan(int x,int y)
{
    int a,b,c,d;

    outp(888,0);
    /* Lectura de las tarjeta PCI-720, con asignación de cada byte a las variables a, b, c, d */

    a=inp(0x2A0+0);
    b=inp(0x2A0+1);
    c=inp(0x2A0+2);
    d=inp(0x2A0+3);

    destino(x,y); /* manda a ejecutar la rutina destino*/

    /* programación del timer contienpo del monoestable en nivel bajo de 4 seg, tiempo suficiente para ir*/
    /* de una posición a otra posición alejada como maximo a 3 lugares de distancia de la posición actual*/

    outp(0x2A0+7,50); /* Byte de control,contador 0,carga LSB y MSB*/
    /* en modo 1, BCD 0 */

    outp(0x2A0+4,0xF4); /* carga LSB */
    outp(0x2A0+4,0x7E); /* carga MSB, 2000 pulsos * 2 mseg = 4 seg */
    outp(890,11); /* genera un flanco positivo en el bus de control */
    outp(890,10); /* del puerto paralelo para disparar el timer*/

    delay((clock_t)2); /* retraso para que se pueda dar el cambio de nivel*/

    /* bloque para identificar el cuadrante a donde se movera*/
    if (a == 0x00)
    {
        if (c == 0x00)
        {
            cuadrante_3(b,d,x,y,cuadx,cuady);
        }
        else
        {
            cuadrante_4(b,c,x,y,cuadx,cuady);
        }
    }
    else
    {
        if(d==0x00)
        {
            cuadrante_1(a,c,x,y,cuadx,cuady);
        }
        else
        {
            cuadrante_2(a,d,x,y,cuadx,cuady);
        }
    }
}

/* Rutina para conocer el destino al que tendra que ir la sonda*/
void destino(int x, int y)
{
    if (x<=8)
    {
        if(y<=8)

```

```

    {
        cuadx=0;
        cuady=2;
    }
    else
    {
        cuadx=0;
        cuady=3;
    }
}
else
{
    if(y<=8)
    {
        cuadx=1;
        cuady=2;
    }
    else
    {
        cuadx=1;
        cuady=3;
    }
}
}
}

```

*/*sirve para poner la sonda en la coordenada indicada, b y d contienen los bytes de la posición dentro de estos segmentos del plano, las variables x, y son la coordenada deseada, cuadx , cuady contienen un valor comprendido entre el 0 y el 3 que determinan los registros de direcciones a leerse para encontrar la posición deseada, finalmente se selecciona una de cuatro acciones a realizarse, aquí unicamente se detecta el movimiento dentro del cuadrante 3*/*

```
void cuadrante_3(int b,int d,int x,int y,int cuadx,int cuady)
```

```

{
    int pointx,pointy,m;
    m=0;
    l=0;
    m=9;
    l=9;
    pointx=cquis[1];
    pointy=yc[1];

    while(pointx != b)
    {
        m++;
        b=inp(0x2A0+1);
        pointx=cquis[m];
    }

    while(pointy != d)
    {
        l++;
        d=inp(0x2A0+3);
        pointy=yc[l];
    }

    if(x>=9)
    {
        if(y>=9)
        {
            if(x>=m)
            {
                if(y>=l)
                {
                    accion_4(x,y,cuadx,cuady);
                }
                else
                {
                    accion_3(x,y,cuadx,cuady);
                }
            }
        }
    }
    else

```

```

        {
            If(y>=t)
            {
                accion_2(x,y,cuadx,cuady);
            }
            else
            {
                accion_1(x,y,cuadx,cuady);
            }
        }
    }
else
{
    If(x>=m)
    {
        If(y>=l)
        {
            accion_4(x,y,cuadx,cuady);
        }
        else
        {
            accion_3(x,y,cuadx,cuady);
        }
    }
    else
    {
        If(y>=l)
        {
            accion_2(x,y,cuadx,cuady);
        }
        else
        {
            accion_1(x,y,cuadx,cuady);
        }
    }
}
else
{
    If(y>=9)
    {
        If(x>=m)
        {
            If(y>=l)
            {
                accion_4(x,y,cuadx,cuady);
            }
            else
            {
                accion_3(x,y,cuadx,cuady);
            }
        }
        else
        {
            If(y>=l)
            {
                accion_2(x,y,cuadx,cuady);
            }
            else
            {
                accion_1(x,y,cuadx,cuady);
            }
        }
    }
    else
    {
        accion_1(x,y,cuadx,cuady);
    }
}
}
}

```

/*Rutina para activar al mot1 atras y al mot2 abajo, se detecta cuando se ha conseguido la posición dentro de un tiempo no mayor a 4 seg, entrega una variable llamada monitor con un cierto valor para que sea interpretada la acción como válida o no válida, funciona de manera identica al programa SETROB.C*/

```
void accion_1(int x,int y,int cuadx,int cuady)
{
  int auxx,auxy;
  int mon;
  outp(888,0);
  auxx=inp(0x2A0+cuadx);
  auxy=inp(0x2A0+cuady);
  mon=inp(889);
  while(auxx != equis[x] || auxy != ye[y])
  {
    auxx=inp(0x2A0+cuadx);
    auxy=inp(0x2A0+cuady);

    if(mon == 0x87)
    {
      /* */
      delay((clock_0)100);
      auxx=inp(0x2A0+cuadx);
      auxy=inp(0x2A0+cuady);
      if(auxy != ye[y])
      {
        outp(888,4);
        if(auxx != equis[x])
        {
          outp(888,5);
          auxx=inp(0x2A0+cuadx);
          auxy=inp(0x2A0+cuady);
        }
        else
        {
          outp(888,4);
          auxx=inp(0x2A0+cuadx);
          auxy=inp(0x2A0+cuady);
        }
      }
    }
    else
    {
      if(auxx != equis[x])
      {
        outp(888,1);
        auxx=inp(0x2A0+cuadx);
        auxy=inp(0x2A0+cuady);
      }
      else
      {
        outp(888,0);
        auxx=inp(0x2A0+cuadx);
        auxy=inp(0x2A0+cuady);
      }
    }
    auxx=inp(0x2A0+cuadx);
    auxy=inp(0x2A0+cuady);
    mon=inp(889);
  } /* */
  else
  {
    outp(888,0);
    monitor=2;
    break;
  }
  mon=inp(889);
}
}
/*+++++++Rutina que proporciona retrasos en milisegundos+++++++*/
void delay(clock_t wait)
{
  clock_t goal;
  goal =wait + clock();
  while (goal > clock() );
}
```

APENDICE B

La interface de comunicación IEEE-488, consiste de un bus que se utiliza fundamentalmente en la interconexión entre uno o varios procesadores y periféricos inteligentes (hasta 15), en este protocolo la transmisión opera en forma asincrónica a razón de 1 Mbyte por segundo, el bus puede extenderse hasta 20 m, con cargas cada 2 m. El bus debe contar con un dispositivo que actuará de controlador, encargado de gestionar las peticiones de uso del bus, ya sea para emitir mensajes (locutores ó talkers), ya para recibirlos (oyentes ó listeners). La figura 4.1 muestra la arquitectura al bus.

Esta descripción pretende aclarar los conceptos necesarios para comprender los fundamentos del funcionamiento del bus.

Señales eléctricas .

El bus consta de 16 señales activas, divididas en tres grupos:

- Bus de datos (8 líneas).
- Bus de control de transferencia de datos (3 líneas).
- Bus para el control general de la interconexión (5 líneas).

El **bus de datos** es un conjunto de ocho líneas de datos, que permite transmitir por el bus, de octeto en octeto, transmite tanto datos, como mensajes de comando. La denominación de las líneas es:

DIO1 (Data input output 1): Dato de entrada o salida 1.

-
-
-

DIO8 (Data input output 8): Dato de entrada o salida 8.

El **bus de control de transferencia**, consta de tres señales, usadas para efectuar la transferencia de cada octeto entre el locutor y el oyente a través de las señales DIO. La denominación y descripción de estas señales es la siguiente:

- DAV (Data Valid): Dato válido. Es emitida por el locutor de la transferencia, e indica que los datos DIO están estables en el bus.
- NRFD (Not Ready For Data): No listo para dato. Es emitida por el oyente de la transferencia, e indica que aún no está listo para recibir nuevos datos.
- NDAC (Not data accepted): Dato no aceptado. Es emitida por el oyente de la transferencia, e indica al locutor que debe mantener los datos en el bus, por que aún no han sido almacenados.

El **bus de control general de la interconexión**, comprende cinco señales, que son empleadas para mantener un flujo ordenado de información a través del bus. Son las siguientes:

- **ATN (Attention):** Atención. Es empleada por el dispositivo que hace las funciones de controlador del bus e indica a todas las demás que está enviando un mensaje de interés general
- **IFC (Interface clear):** limpiar la interconexión. el controlador indica al resto de los dispositivos que deben volver al estado inicial (reset), o reposo.
- **SRQ (Service Request):** Petición de servicio. Los dispositivos no controladores usan esta línea para indicar al controlador sus deseos de utilizar el bus para efectuar una transferencia de datos.
- **REN (Remote Enable):** Activa el control remoto. El control indica a los dispositivos direccionados que deben ignorar el control local, panel frontal o similar, para obedecer al control remoto recibido a través del bus.
- **EOI (End or Identify):** Fin o identificación. Puede ser activada por el dispositivo locutor o por el controlador. En el primer caso indica el fin de la transmisión de un bloque de datos. En el segundo caso, el controlador indica a los dispositivos que han pedido servicio que se identifiquen.

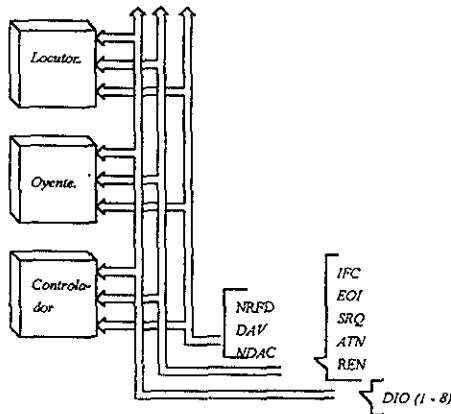


Fig. 4.1 Arquitectura de la interfaz IEEE-488.

Especificaciones del conector.

El tipo de conector que se usa es el recomendado en la norma IEEE, es un conector de 24 contactos como se ve en la figura 4.2. El estándar de los conectores es el Amphenol o Cinch Serie 57 Microribbon o tipo Amp Champ



Fig. 4.2 Conector empleado.

Como se puede ver en la figura 4.2, el conector es de forma trapezoidal el cual distribuye las señales como se indica en la siguiente tabla 4.1.

Todas las señales se transmiten por el bus con lógica negativa (1=Bajo ≤ 0.8 V ; 0=Alto ≥ 2.0 V).

Contacto.	Señal de la línea	Contacto.	Señal de la línea.
1	DIO1	13	DIO5
2	DIO2	14	DIO6
3	DIO3	15	DIO7
4	DIO4	16	DIO8
5	EOI	17	REN
6	DAV	18	Gnd,(6)
7	NRFD	19	Gnd,(7)
8	NDAC	20	Gnd,(8)
9	IFC	21	Gnd,(9)
10	SRQ	22	Gnd,(10)
11	ATN	23	Gnd,(11)
12	SHIELD	24	Gnd,LOGIC

Tabla 4.1 Asignación de las líneas del bus a las terminales del conector. Gnd, (n) se refiere a la señal de retorno del contacto 'n'.

Funciones a realizar por los dispositivos conectables al bus.

Para que un dispositivo sea capaz de interpretar una señal o conjunto de ellas es preciso que se le haya dotado de los circuitos necesarios para analizarlos y responder han función de dichas señales. Estos circuitos activarán o no unas señales, locales del dispositivo o generales del bus, en función de las señales que interpreten.

Según esto, un dispositivo se considera descompuesto en diez funciones distintas, cada una de las cuales es relativamente independiente de las demás y se puede analizar separadamente. Las funciones que puede realizar un dispositivo conectado al bus son las indicadas en la tabla 4.2.

Función de la interfaz.	Símbolo.
Protocolo surtidor.	SH
Protocolo receptor	AH
Locutor o locutor extendido.	T ó TE
Oyente u oyente extendido.	L ó LE
Petición de servicio.	SR
Control remoto o local.	RL
Encuesta paralelo.	PP
Borrado del dispositivo.	DC
Disparo del dispositivo.	DT
Controlador.	C

Tabla 4.2 Repertorio de funciones de la interfaz.

Controlador (Controller = C). Es el dispositivo que realiza las funciones de controlador, encargado de inicializar todas las comunicaciones, que se establezcan a través del bus. El rol del controlador puede ser comparado con el rol del CPU de una computadora, Cabe mencionar que una computadora puede ser el dispositivo controlador del bus.

En el bus debe haber siempre un dispositivo que reciba las peticiones de uso del mismo, ya sean de un dispositivo que desea emitir un mensaje, ya sea de un que espera recibirlo. Este dispositivo analizará las peticiones de servicio, averiguando el dispositivo o dispositivos que las han realizado y autorizándoles al uso del bus por orden de prioridad, según los criterios con los que se haya programado el controlador.

El controlador tiene además la capacidad de enviar órdenes a todos los dispositivos conectados al bus, modificando el estado interno de éstos. También activa la señal ATN, para indicar a los demás dispositivos que está enviando por el bus un mensaje de interés general.

EN un momento dado, solo puede haber un dispositivo ejerciendo las funciones de controlador: El resto de los dispositivos capaces de ejercer esta función deben mantenerla en estado inactivo.

Locutor (Talker = T ó TE). La función de locutor permite a un dispositivo transmitir información a otros dispositivos conectados al bus. Un dispositivo con capacidad para realizar esta función sólo podrá ejercerla cuando haya recibido del controlador la orden de transmitir un mensaje, es decir, cuando el controlador lo haya seleccionado como locutor del bus.

Todo dispositivo capaz de actuar como locutor tiene asignada una dirección, que le diferencia de los otros dispositivos conectados al bus con esta función. Esta dirección se llama 'dirección del locutor' y puede transmitirse en uno o dos octetos. En el segundo caso el proceso de direccionado del dispositivo se hace en dos etapas, y la función del locutor se complica, recibiendo el nombre de locutor extendido (Talker Extended = TE).

Oyente (Listener = L ó LE). Esta función capacita al dispositivo que la ejercita para recibir mensajes a través del bus. Estos mensajes serán transmitidos por el dispositivo que esté ejerciendo en ese momento la función de locutor.

La función de oyente sólo puede ser ejercida por los dispositivos, que han sido seleccionados por el controlador como oyentes del bus. Lo mismo que en el caso del locutor, esta selección la hace el controlador enviando por el bus la 'dirección del oyente', puede transmitirse en uno o dos octetos. En el segundo caso la función que permite al dispositivo reconocerse como oyente del bus se llama oyente extendido (Listener Extended = LE).

Protocolo surtidor (Source Handshake = SH). Esta función permite al dispositivo enviar un dato cualquiera por el bus, ya sea parte de un mensaje, o una respuesta a una orden del controlador. La función de locutor solamente capacita al dispositivo para adquirir el control del bus de datos, como dispositivo emisor de mensajes

Se requiere de esta función para controlar el protocolo de las señales, que asegura la llegada del dato transmitido a todos los dispositivos que han sido seleccionados para aceptarlo.

Protocolo receptor (Acceptor Handshake = AH). Permite al dispositivo recibir los datos transmitidos por el bus. Esta función está directamente relacionada con el protocolo del surtidor de mensaje, dado que la transferencia de datos es asincrónica, y que puede haber más de un dispositivo recibiendo el dato, el protocolo receptor permite al dispositivo retardar el inicio de la transmisión del dato, indicando que uno está listo (NRFD), y retardar el final de la transferencia de un dato, indicando que aún no lo ha aceptado o recibido (NDAC).

Petición de servicio (Service Request = SR). Permite, al dispositivo dotado de esta función, solicitar asincrónicamente del controlador el uso del bus, para esto activa la señal SR del bus, y la mantiene en este estado hasta que el controlador indica al dispositivo que ha recibido la solicitud del servicio.

Encuesta en paralelo (Parallel Poll = PP). Esta función proporciona al dispositivo la capacidad de presentar un bit de estado al controlador sin haber sido direccionado como locutor. Esta capacidad es ejercida por el dispositivo cuando el controlador hace una encuesta en paralelo, preguntando a una serie de dispositivos pre seleccionados si requieren o no el servicio de bus. Esto permite al controlador discernir entre ocho dispositivos, o grupos de dispositivos, uno por cada línea de datos, con una sola pregunta.

Borrado del dispositivo (Device Clear = DC). Permite borrar ó inicializar, al dispositivo, individualmente o como parte de un grupo de ellos. El grupo puede ser un subconjunto o la totalidad de los dispositivos direccionables conectados al bus.

Es necesario diferenciar esta función DC y la señal IFC del bus. La función DC inicializa el dispositivo, llevándolo al estado inicial para el que ha sido programado, la señal IFC, inicializa las funciones de interconexión con el bus y no afecta al dispositivo conectado al bus a través de estas funciones.

Disparo del dispositivo (Device Trigger = DT). Esta función permite poner en marcha las funciones básicas del dispositivo, individualmente o en grupo.

Control remoto o local (Remote Local = RL). Permite seleccionar entre dos fuentes de información de entrada, La función indica al dispositivo si debe usar la información recibida del panel frontal, local, o la recibida a través de la interconexión del bus, en forma remota.

Mensajes transmitidos por el bus.

En seguida se dan a conocer los distintos mensajes transmitidos por el bus, junto con las señales que los componen

Atención ATN. Este mensaje consiste en activar la señal de atención del bus, (esta operación solo puede ser efectuada por el dispositivo controlador del bus). Cuando el controlador está enviando este mensaje, todos los demás dispositivos deben permanecer en silencio. El mensaje indica a todos los dispositivos conectados al bus que el controlador está enviando por el bus de datos un mensaje que puede ser de interés para cualquier dispositivo.

Mandatos universales UCG (Universal Command Group). Son los mandatos que deben ser ejecutados por todos los dispositivos capaces de interpretarlos . Estos mandatos son ocho y son los siguientes:

- Limpiar dispositivo: **DCL** .Activa la función de borrar el dispositivo (DC).
- Bloqueo local: **LLO** (Local Lockout). Al recibir la función RL este mensaje, impide que el dispositivo pueda cambiar de control remoto a control local, como consecuencia de una orden local.
- Desmontar la encuesta paralelo: **PPU** (Parallel Poll Unconfigure). Retira el permiso para identificarse como peticionario del servicio del bus, cuando el controlador hace una encuesta en paralelo, a todos los dispositivos que lo tenían en ese momento.
- Inicio de encuesta serie: **SPD** (Serial Poll Enable). Indica a todos los dispositivos que les va a preguntar su estado por turno, uno tras otro, para determinar si han solicitado el servicio del bus.
- Fin de encuesta serie: **SPD** (Serial Poll Disable). Indica a todos los dispositivos que han terminado de preguntar los estados internos.
- Identificación: **IDY** (Identify). Este mensaje lo transmite el controlador activando las señales de EOI y ATN al unísono. Indica la realización de una encuesta paralelo. Deben responder todos los dispositivos a los que se ha validado la facultad de hacerlo, indicando si solicitan o no el servicio del bus.
- Limpiar la interconexión: **IFC** (Interface Clear). Este mensaje inicializa todas las funciones de la interfaz conectadas al bus.
- Validar control remoto: **REN** (Remoto Enable). Este mensaje se debe enviar junto a la dirección de oyente del dispositivo al que se quiere dar la orden de actuar en función de los mensajes recibidos a través del bus. También se puede enviar junto al de bloqueo local, lo que provoca al paso de todos los dispositivos a un estado en que sólo el controlador puede ordenarles el cambio de control remoto a local, o viceversa.

Mandatos selectivos ACG (Addressed Command Group) Son los mensajes enviados por el controlador que sólo pueden ser interpretados por él o los dispositivos que han sido seleccionados previamente para hacerlo. Los mandatos selectivos son los siguientes:

- Disparo de un grupo: **GET** (Group Execute Trigger). Pone en marcha las operaciones básicas de los dispositivos previamente direccionados como oyentes
- Pasar a local: **GTL** (Go to Local). Deja bajo control local a los dispositivos previamente direccionados como oyentes

- Montar la encuesta paralelo: **PPC** (Parallel Poll Configure). Indica, al o a los dispositivos que están seleccionados como oyentes, que se les va a autorizar o desautorizar para responder cuando se haga una encuesta paralelo.
- Limpiar los dispositivos seleccionados: **SDC** (Selected Device Clear). Inicializa (**DC**) los dispositivos previamente seleccionados como oyentes.
- Toma el control: **TCT** (Take control). El controlador envía este mensaje para transferir el control del bus a otro dispositivo, que habrá sido seleccionado previamente como locutor.

Direcciones: **AD**. Para seleccionar un dispositivo como locutor o como oyente, ya sea para inicializar un diálogo, o para transmitirle a continuación un mandato selectivo, se emplean los mensajes de dirección.

Todos los dispositivos que tienen la función que les permite actuar de locutores tienen una dirección de locutor distinta. Todos los que tienen la función de oyente tienen su propia dirección de oyente. Los dispositivos, que tienen las dos funciones tienen la misma dirección en ambas. El controlador puede enviar cuatro mensajes de dirección distintos:

- Dirección de oyente: **MLA** (My Listen Address). Cada uno de los dispositivos conectados al bus detectan, por los bits 6 y 7 de los datos, que la dirección es de oyente. La dirección propiamente dicha la constituyen los bits 1 a 5 del bus de datos. Puede haber más de un dispositivo con la misma dirección de oyente.
- No oír: **UNL** (Unlisten). La dirección de oyente u oyente más alta de los posibles, 11111, no corresponde a ningún dispositivo, y es interpretada por todos los dispositivos, que estaban seleccionados como oyentes, como una orden para que dejen de atender a los mensajes transmitidos por el bus.
- Dirección de locutor: **MTA** u **OTA**. Cuando el controlador envía por el bus una dirección de locutor o locutor, el dispositivo al que corresponde esa dirección queda seleccionado como locutor para futuras comunicaciones a través del bus. El dispositivo que estaba actuando como locutor detecta que se ha transmitido otra dirección de locutor y deja de actuar como tal. Un mismo mensaje de dirección de locutor es interpretado de dos formas distintas por el dispositivo que actuaba de locutor y por el propietario de la dirección transmitida, estas dos interpretaciones tienen las siglas:
 - MTA**: Mi dirección de locutor (My Talk Address).
 - OTA**: Otra dirección de locutor (Other Talk Address).
- No hablar: **UNT** (Untalk). La dirección 31 es interpretada por los dispositivos capaces de hablar (actuar de locutor), como una orden para dejar de transmitir mensajes

Mensajes secundarios: **SR**. Son los mensajes que complementan a otro enviado previamente. Son los siguientes:

- Dirección secundaria: **MSA** u **OSA**. Al hablar de las funciones **T** y **L** (locutor y oyente), se apuntaba que la dirección podría ser transmitida en uno o dos octetos. La dirección secundaria constituye el segundo octeto de una dirección de locutor o de oyente.

- Validación de encuesta paralelo: **PPE** (Parallel Poll Enable). El mensaje universal de asignación de encuesta paralelo (**PPC**) va seguido de un permiso para responder a la encuesta paralelo, para el último dispositivo que ha sido direccionado como oyente. A este dispositivo se le asigna uno de los ocho bits de datos, para que indique si requiere o no el servicio del bus cuando se haga la encuesta paralelo.
- Invalidar encuesta paralelo: **PPD** (Parallel Poll Diable). Para poder asignar un bit del bus de datos a un dispositivo es necesario haberlo des asignado al que lo usaba hasta el momento. Para ello se pueden usar dos mandatos, el **PPU**, que desasigna todos los bits al mismo tiempo, y el **PPD**, que desasigna sólo los bits de los dispositivos actualmente direccionados como oyentes.

Mensajes de estado: **ST**. Son los mensajes enviados por los dispositivos para indicar su estado interno.

- Fin: **END**. Indica el final de una transmisión de datos entre locutor y uno o varios dispositivos oyentes.
- Octeto de estado: **STB** o **RQS**. Cuando el controlador hace una encuesta en serie, al interrogar a un dispositivo éste responde enviando por el bus de datos una palabra de estado. En general este mensaje recibe el nombre de octeto de estado.
- Petición de servicio: **SRQ** (Service request). Se envía forzando a una línea especial, **SRQ**, en paralelo con el mensaje que se esté enviando en ese instante.
- Respuesta a encuesta paralelo: **PPRi**. Cuando el controlador envía el mensaje **IDY**, indicando la realización de una encuesta paralelo, cada uno de los ocho dispositivos asignados responden con uno de estos mensajes, colocando el bit correspondiente a 1 si requieren el uso del bus. Dado que cada dispositivo sólo actúa sobre uno de los ocho bits del bus de datos, los ocho mensajes se pueden enviar al mismo tiempo, razón por la cual se dice que la encuesta es en paralelo.

Mensajes de manipulación de datos: **HS** (Handshake). Estos mensajes son los empleados por las funciones **AH** y **SH** para sincronizarse cada vez que la segunda envía un octeto a la primera

- Dato aceptado: **DAC** (Data accepted). La función **AH** del dispositivo oyente indica que ya ha almacenado el octeto que se le está enviando por el bus.
- Dato válido **DAV** (Data valid). La función **SH** del dispositivo locutor indica que los bits enviados por el bus de datos ya están estables y que pueden ser almacenados por los oyentes..
- Listo para dato: **RFD** (Ready For Data). la función **AH** del oyente indica que ya ha tratado el último dato almacenado y que puede tomar un nuevo octeto del bus de datos.

Mensajes dependientes del dispositivo. **DD** (Device Dependent) Son los mensajes de datos (**DAB** = Data byte) que se intercambian el locutor y el oyente, una vez que el controlador les ha transferido el servicio del bus. Se distinguen dos casos particulares.

- Fin de una cadena de datos: **EOS** (End of String), que es el código interpretado por los dos dispositivos como fin de diálogo
- Nulo: **NUL**. Es un octeto con ceros.

Transferencia de datos.

La transferencia de datos se realiza a través de las líneas DIOi. la sincronización entre emisor y receptor se consigue con las señales DAV, NRFD y NDAC, tal como ya se ha indicado. El protocolo de la transferencia se realiza de acuerdo con el cronograma de la figura 4.3.

Cuando el locutor ha situado el dato en el bus, lo indica enviando el mensaje DAV (dato válido). los oyentes, al detectarlo, dejan de enviar el mensaje RFD (listo para dato) y, tras tomarse el tiempo necesario, van indicando que han aceptado el dato, enviando el mensaje DAC (dato aceptado). El bus realiza una operación AND-cableada con esta señal, lo que no permite al locutor recibir el mensaje hasta que todos los oyentes lo están enviando.

Cuando el locutor detecta el mensaje DAC enviado por todos los oyentes, retira del bus el mensaje DAV. Al detectarlo los oyentes, retiran el mensaje DAC y vuelven a enviar el mensaje RFD, cuando estén listos para almacenar un nuevo dato.

Para que el locutor detecte el mensaje RFD deben enviarlo todos los oyentes, pues el bus hace una operación AND cableada con esta señal. Cuando el locutor recibe el mensaje RFD, se inicia nuevamente el proceso.

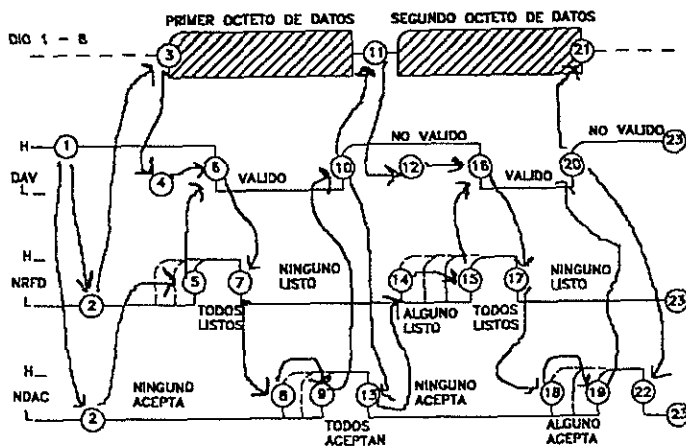


Figura 4.3 Cronograma de las señales empleadas en el protocolo de manipulación de datos cuando hay un locutor y varios oyentes.

Petición de servicio del bus (interrupción).

Cuando un dispositivo pone activa (1 = baja) la línea SRQ, el controlador interpreta que algún dispositivo está deseando utilizar el bus.

Como puede haber más de un dispositivo solicitando el bus en un instante determinado, el bus debe hacer una operación OR con todos los mensajes (señales) SRQ,

de modo que cuando una o varias de ellas estén activas (bajas) el controlador recibe una señal activa (baja).

De acuerdo a lo anterior existen actualmente tarjetas conocidas como GPIB's (General Purpose Interface Bus, bus de interconexión de propósito general) que utilizan el estandar IEEE-488, descrito anteriormente, y pueden ser insertadas en el bus interno de cualquier computadora personal aprovechando todas sus capacidades.

APENDICE C

La tarjeta NI - 488.2, es una tarjeta que cumple con los requerimientos de la interfaz IEEE-488 (GPIB). se utilizó en el proyecto por que dentro de la sección de acústica del centro de instrumentos, ya se contaba con ella aun que sabemos que en el mercado se tienen tarjetas GPIB similares de fabricantes como son: Motorola, Intel, Hewlet Pakard, etc.

En realidad NI - 488.2, es un conjunto de dos elementos, por un lado se tiene la tarjeta con todos los dispositivos electrónicos o hardware, y el software o programas capaces de controlar a esta tarjeta. Es de la marca National Instruments, y dentro del paquete solo incluye manuales de instalación del software y no proporciona mayor información sobre la electrónica de su tarjeta, razón por la cual se describe en este capítulo la forma de instalación del software, configuración y operación con algunas rutinas y funciones.

Al termino NI - 488.2 , tenemos que agregarle para MS - DOS, (NI - 488.2 para MS - DOS) es el nombre que se le da al producto que contiene a la tarjeta de interfaz que es 100% compatible con el estandar IEEE - 488, y al software usado específicamente para el control de la tarjeta, bajo un ambiente de MS - DOS. Por que 488.2, se debe a que es el estándar IEEE - 488.2, es decir la última revisión efectuada en 1987.

Antes de pasar en pleno al manejo, cabe mencionar que, la tarjeta NI - 488.2 esta soportada por manejadores para MS-DOS y también para Windows aun que este proyecto esta en plataforma de MS-DOS.

Instalación y configuración.

Antes que nada se tiene que mencionar que la computadora empleada para este proyecto utiliza una tarjeta madre con procesador Intel 486 y velocidad de 66 Mhz. El paquete contiene en el disco que es distribuido los siguientes programas y archivos, requeridos para poder programar a la GPIB en los siguientes lenguajes BASICA, Quick Basic, Microsoft BASIC, o Microsoft C:

- GPIB.COM Es el manejador del NI-488.2 que es cargado al inicializar al sistema
- BIB.M Es un archivo de interconexión, que permite el acceso de los programas de aplicación hechos en lenguaje BASICA, a el software NI-488.2
- QBIB.OBJ Es un archivo de interconexión, que permite el acceso de los programas de aplicación hechos en lenguaje QuickBASIC (versión 4.0 o mayores), a el software NI-488.2.
- MBIB.OBJ Es un archivo de interconexión, que permite el acceso de los programas de aplicación hechos en Microsoft Professional BASIC (versión 7.0), a el software NI-488.2.
- MCIB.OBJ Es un archivo de interconexión, que permite el acceso de los programas de aplicación hechos en Microsoft C (versión 4.0 o mayores), a el software NI-488.2.
- DECL.BAS Es un archivo que contiene los códigos de declaración, que se deben de poner al comienzo de los programas de aplicación hechos en BASICA
- QBDECL.BAS Es un archivo que contiene los códigos de declaración, que se deben de poner al comienzo de los programas de aplicación hechos en QuickBASIC
- MBDECL.BAS Es un archivo que contiene los códigos de declaración, que se deben de poner al comienzo de los programas de aplicación hechos en Microsoft BASIC
- DECL.H Es un archivo de encabezado que se usa en las aplicaciones de Microsoft C

Adicionalmente se proporcionan otros programas y archivos, que son empleados para instalación y prueba del software, diagnostico de problemas en la tarjeta, y ejemplos de programas de aplicación. Dichos programas y archivos son los siguientes:

- **IBDIAG.EXE** Es un programa que se usa para probar el adecuado funcionamiento de la tarjeta.
- **IBTEST.EXE** Es un programa que se usa para probar el software.
- **INSTALL.EXE** Es un programa con manejador de menú, que se usa para instalar el software NI-488.2 y probar la tarjeta, así como configurar a dicho software.
- **IBCONF.EXE** es un programa de configuración del software, que puede ser usado para cambiar la configuración del NI-488.2.
- **IBIC.EXE** Es un programa de control interactivo que ejecuta funciones del NI-488.2, que son dadas des de el teclado. Ayuda a comprender las funciones, programar instrumentos o dispositivos GPIB, y desarrollar programas de aplicación.
- **APPMON.EXE** Es un programa de monitoreo de las aplicaciones. Se usa para depurar las aplicaciones.
- **IBTRAP.EXE** Es un programa que configura las aplicaciones monitor.
- **ULI.COM** Es un archivo necesario para usar la opción de, interfaz de lenguaje universal
- **DBSAMP.BAS, DQBSAMP.BAS, DCSAMP.C** y **DIBSAMP** Son ejemplos de programas con llamas a los dispositivos, hechos en, BASICA, QuickBASIC, BASIC, C.
- **BBSAMP.BAS, BQBSAMP.BAS, BMBSAMP.BAS, BCSAMP.C** Son ejemplos de programas para llamadas a las tarjetas, hechos en, BASICA, QuickBASIC, BASIC, C.
- **BSAMP488.BAS, QBSAMP488.BAS, MSAMP488.BAS, CSAMP488.C** Son ejemplo de programas con llamadas 488.2, hechos en, BASICA, QuickBASIC, BASIC, C.

Para instalar y configurar el manejador se hace lo siguiente, en el disco de distribución de la NI-488.2 se tiene un programa llamado **INSTALL.EXE** que instala la programación de la NI-488.2 y prueba la tarjeta y su configuración, dentro de este programa se indican los pasos a seguir para continuar con la instalación, como ejemplo se indica que debe de modificarse el archivo de configuración del sistema MS-DOS, y que es el **CONFIG.SYS**, agregándole la siguiente línea:

```
device=dir\gpiib.com
```

donde **dir** es el directorio en el cual esta el software de la NI-488.2 .

Dentro de este proyecto y después de realizada la instalación se utilizó el programa de configuración, lo cual se realiza por medio del programa llamado **IBCONF.EXE**, que es un programa que se usa para configurar las características de la tarjeta GPIB y de los dispositivos conectados a está; mediante este programa se le indicó a la NI-488.2 los dispositivos y sus direcciones que controlaría, así como la propia dirección de la GPIB, también se configuraron las características del dispositivo conectado a la GPIB que en este caso es el analizador de frecuencias 2133 de la marca B&K.

El programa **IBCONF.EXE** proporciona un ambiente gráfico, consistente de ventanas en las cuales se describen las tarjetas y dispositivos que hacen uso del sistema Este programa puede soportar hasta un total de cuatro tarjetas GPIB, a de más de 32 dispositivos, es decir que se puede tener a las GPIB0, GPIB2,GPIB3 y GBIB4 Dentro de las opciones que permite el programa están:

- 1 - Mapas de los dispositivos en las tarjetas.
- 2 - Ayuda.
- 3.- Renombrar.
- 4 - (Des)conectar

- 5 - Editar.
- 6.- Abertura de la configuración de la GPIB.
- 7 - Autoconfiguración.
- 8.- Salida.

En la primera opción, Mapas de los dispositivos en las tarjetas, es posible ver los dispositivos asignados a cada tarjeta GPIB, por omisión todas las tarjetas GPIB, se les asigna el mismo número de dispositivos.

En la segunda opción, Ayuda, Se tiene acceso a la información que nos ayuda a comprender el uso del programa IBCONF.EXE.

En la opción, Renombrar, es posible cambiar el nombre a los dispositivos conectados en las GPIB's. Los nombres de los dispositivos, como máximo emplearan ocho caracteres.

La opción, (Des)conectar, se usa para conectar o desconectar algún dispositivo de una tarjeta.

En la opción, Editar, se usa para editar o examinar las características particulares de alguna tarjeta o de algún dispositivo.

La opción, Abertura de la configuración de la GPIB, cuando se configura un manejador GPIB, una versión de esta configuración puede ser escrita en modo texto en un archivo del disco, este archivo es solo para propósito de información.

La opción, Autoconfiguración, realiza una configuración automática, lo que realiza esta opción, es preguntar por todos los dispositivos oyentes (listeners), para realizar un ajuste en el mapa de dispositivos de la tarjeta para que solo los dispositivos que estén conectados respondan.

Por último la opción Exit, se usa para dejar al programa IBCONF, pero no sin antes de ser cuestionados si se desea o no guardar los cambios hechos en las configuraciones.

Para la configuración de la tarjeta y de los dispositivos conectados a esta, el programa IBCONF despliega una ventana en la que se definen las características, direcciones e interrupciones, la configuración de dispositivos especifica las características usadas para el acceso a la tarjeta cuando las funciones del dispositivo son usadas, la configuración de las tarjetas especifica las características usadas cuando las funciones de la tarjeta son requeridas.

Las siguientes funciones son disponibles:

- Cambio de características.
- Cambio de tarjeta o dispositivo.
- Ayuda.
- Valor de reinicialización
- Regreso al mapa.

Cambio de características. Es para cambiar alguna característica de un dispositivo o tarjeta.

Cambio de tarjeta o dispositivo Es para cambiar a la siguiente tarjeta o dispositivo, o bien a la anterior

Ayuda. Proporciona ayuda para comprender las características de la configuración.

Valor de reinicialización. Se usa para reinicializar alguna característica

Regreso al mapa. Regresa a una de las pantallas en las que se tiene una gráfica con la tarjeta GPIB y sus dispositivos conectados con sus valores de omisión o default.

Las configuraciones por omisión o default. pueden ser por ejemplo, que los nombres de los 32 dispositivos son DEV1 hasta DEV32. Aunque se puede renombrar a cada dispositivo para tener una mayor descripción de este, como por ejemplo si tenemos conectado a la GPIB un multimetro digital se le puede poner en la dirección 1, con el nombre de MULT, en lugar de nombrarse DEV1, en el caso concreto de este proyecto el dispositivo empleado (analizador de frecuencias 2133 de B&K) tiene una dirección que corresponde al 17 en hexadecimal, y que tiene el nombre de BK2133.

Los valores de omisión del programa son los siguientes:

- Son 32 dispositivos cuyos nombres simbólicos son DEV1 hasta DEV32.
- Pueden ser 4 tarjetas cuyos nombres de acceso son GPIB0, GPIB2, GPIB3 y GPIB4. Los nombres de acceso de estas tarjetas no puede ser cambiado.
- El acceso a la tarjeta GPIB0 es activada, mientras que el acceso de las otras es desactivado.
- La dirección en la GPIB de los primeros 16 dispositivos es la misma que el número de dispositivo. Por ejemplo, DEV1 esta en la dirección 1.
- Los 16 dispositivos restantes (que son del 17 al 32) son asignados a la tarjeta GPIB1. Donde el rango de dirección va de 1 a la 16, respectivamente, por ejemplo, DEV17 esta en la dirección 1.
- Cada tarjeta de interfaz GPIB es un sistema controlador de su bus, y tiene una dirección de 0.
- El mensaje END es enviado con el último byte de cada mensaje de datos a el dispositivo. El carácter End-of-String (EOS) no es reconocido.
- El tiempo límite de I/O y llamadas a funciones de espera es puesto para aproximadamente 10 segundos
- Cada tarjeta GPIB esta puesta en sus valores de omisión para su dirección base, sus interrupciones, y canales DMA (Direct Memory Access).

La tabla 1 y la tabla 2 muestran la forma en como fue configurada la tarjeta GPIB así como la configuración de las características del analizador de frecuencias 2133, respectivamente. Estas características obedecen a las indicaciones que se dan en los manuales tanto de la GPIB como del analizador, a si por ejemplo la documentación del analizador indica una dirección primaria iguala al 17h, por lo que en donde se requiera se tendrá que indicar que el analizador 2133 tiene una dirección primaria 17. Posteriormente se dará una explicación de que significa cada característica contenida en las dos tablas siguientes.

GPIB0 Configuration.

```

Primary GPIB Address..... 0
Secondary GPIB Address..... NONE
Timeout setting..... 10 sec

Terminate Read on EOS..... Yes
Set EOI with EOS on Writes..... Yes
Type of compare on EOS..... 8-Bit
EOS byte..... 00h
Send EOI at end write..... Yes

System Controller..... Yes
Assert REN when SC..... No
Enable Auto Serial Polling..... Yes
Enable CIC Protocol..... No
  
```

```

Bus Timing . . . . . 500nsec
Parallel Poll Duration... . . . . Default

Use this GPIB board . . . . . Yes
Board type..... . . . . PC11A
Base I/O Address . . . . . 02E1H
DMA Channel . . . . . 1

```

Tabla 1 Configuración de la tarjeta GPIB.

BK2133 Configuration

```

Primary GPIB Address. . . . . 17
Secondary GPIB Address . . . . . NONE
Timeout setting... . . . . 10sec

Terminate read on EOS..... . Yes
Set EOF with EOS on writes..... Yes
Type of compare on EOS . . . . . 8-Bit
EOS byte..... . . . . 00h
Send EOF at end of Write . . . . . Yes

Enable Repeat Addressing..... No

```

Tabla 2 Configuración del analizador de frecuencias 2133 de B&K.

Características de los dispositivos y tarjetas.

Aquí se explican las características de los dispositivos en el programa IBCONF que son comunes a todas las revisiones de la NI-488.2 para MS-DOS, y que se pueden ver como fueron configuradas en las tablas 1 y 2.

- **Primary GPIB Address** Todos los dispositivos y tarjetas son asignados a una dirección primaria única en el rango de 00hex hasta 1Ehex (0 a 30 decimal). El manejador automáticamente forma una dirección de oyente al sumar a la dirección primaria 32hex, y forma la dirección de locutor cuando suma 64hex a la dirección primaria. La dirección primaria de un dispositivo es puesta dentro del propio dispositivo y puede cambiarse por switches o por programa, en las tarjetas GPIB por omisión su dirección primaria es 0 y solo puede cambiarse por programa. En este proyecto la dirección primaria para la GPIB es 0 y para el analizador BK2133 es la 17
- **Secondary GPIB Address.** Cualquier dispositivo o tarjeta que este usando una dirección extendida le debe ser asignado una dirección secundaria, en el rango de 60hex a 7Ehex (96 a 126 decimal), o también la opción NONE o ninguna, que des activa la dirección secundaria para cuando se tienen dispositivos con tan solo dirección primaria. La opción por omisión para esta característica es NONE.
- **Timeout Settings.** Es el mínimo valor de tiempo aproximado a la cantidad de tiempo que toma la ejecución de las funciones de entrada-salida, antes de que ocurra una interrupción. Por omisión el valor es de 10 sec.
- **Terminated READ on EOS** Algunos dispositivos mandan una señal de EOS (End of String) final de cadena. Una respuesta positiva a esta característica le indica a la tarjeta GPIB la terminación de una operación de lectura. La opción por omisión de esta característica es NO. En este proyecto el analizador si manda una señal de EOS por lo que se tiene que poner una respuesta positiva o YES, tanto en la configuración de la GPIB como en la del analizador
- **Set EOF with EOS on Write.** Una respuesta positiva a este campo es cuando la GPIB reconoce la línea EOF (End of Identify, fin o identificación), cuando también esta activa la línea EOS en operación de escritura, entonces el controlador envía un mensaje de fin cuando es enviado el carácter EOS. La opción por omisión es

NO. Debido a que el dispositivo esta mandando una señal de EOS ala GPIB se tienen que configurar a ambos con una respuesta YES.

- **Type of Compare on EOS.** Este campo indica el tipo de comparación para hacerse con el byte de EOS. Se indica si es en ocho bits o solo los 7 bits menos significativos, la opción por omisión es de 7 bits. En el proyecto el manual del analizador indica que es una comparación de 8 bits
- **EOS Byte** Es posible programar algunos dispositivos para terminar una operación de lectura cuando un carácter previamente seleccionado es detectado. El carácter retorno de línea (0A hex) es un byte EOS común en muchas aplicaciones. La opción para esta característica es 00h.
- **Set EOI at End of Write.** Algunos dispositivos, estando como oyentes, requieren que el orador termine un mensaje de datos con la línea de EOI en el ultimo byte. La opción por omisión es YES.
- **System Controller** (característica solo para tarjetas). Sirve para definir si una GPIB es un sistema controlador el cual es el que tiene el mando sobre el bus. En algunas redes de computadoras enlazadas con la interfaz GPIB a algún otro dispositivo puede ser designado como controlador en cuyo caso la respuesta a esta característica es NO. Por omisión la respuesta es YES.
- **Assert REN when SC.** Una respuesta positiva a este campo causa una activación remota (REN) para ser afirmada automáticamente cada que la tarjeta es puesta en línea, previendo antes que el sistema tenga la capacidad para ser sistema controlador. La opción por omisión es NO.
- **Enable Auto Serial Polling.** Esta opción activa o desactiva la encuesta serial de dispositivos cuando la línea SRQ (Service Request, petición de servicio) es activada. La opción por omisión es NO En el proyecto es necesaria una encuesta serial de dispositivos por lo que la opción es activada con un YES.
- **Enable CIC protocol.** Si un dispositivo hace una llamada después de que el control ha sido pasado a otro dispositivo, activando este protocolo causa que la tarjeta reconozca un SRQ con una encuesta serial. Lo que hace es que el controlador actual reconozca que la tarjeta busca recuperar el control, si la tarjeta actual lo reconoce pasa el control a la tarjeta anterior. Por omisión es YES. En esta opción se configuro con un NO, por que solo se tiene conectada una tarjeta GPIB al bus.
- **Bus Timing.** Este campo especifica el retraso para que se pueda manipular datos de una fuente. Entonces es retraso con la minima cantidad de tiempo, para que la tarjeta capte el dato, después de que a sido puesto en el bus. Si la longitud total de cable de la GPIB es menor a 15 m, el valor es de 350 nsec. La opción por omisión es de 500 nsec.
- **Parallel Poll Duration.** Este campo indica la longitud de tiempo que espera el manejador para conducir una encuesta paralelo. Para una configuración normal la duración de default es correcta.
- **Use This Board.** Es solo si se quiere que el controlador no tenga un acceso a la tarjeta. Se tiene que seleccionar YES.
- **Base I/O Address.** Este campo especifica la dirección de I/O en la tarjeta GPIB. La dirección dada por el fabricante es 02E1h.
- **DMA Channel.** Este campo especifica el canal de DMA (Direct Memory Access, acceso directo a memoria) a ser empleado por la GPIB. El fabricante da el canal 1

Forma de programar en Microsoft C.

Una vez configurados los parámetros y la tarjeta GPIB, es posible desarrollar programas de aplicación en los siguientes lenguajes: BASICA, Quick BASIC, BASIC y lenguaje C (especificamente de MICROSOFT, razón por la que no es posible programar en Borland C por ejemplo, debido a la compatibilidad) El software proporcionado, consiste, como ya se menciona, de manejadores de alta velocidad y varias utilerías para el desarrollo y depuración de programas de aplicación. El software del NI-488.2 se puede acceder en tres formas: directamente con las rutinas NI-488.2, con funciones NI-488.2, o por caracteres de baja ejecución que utilizan comandos estilo Hewlett Packard.

Este proyecto se especifico que sería programado en lenguaje Microsoft C ver. 7 el software del disco de distribución, contiene los siguientes archivos necesarios para poder programar en Microsoft C, versión 5.0 o posteriores, y son:

DECL.H es un archivo que contiene variables útiles y declaraciones constantes
MCIB OBJ es la interfaz entre los programas de aplicación y el acceso a el controlador.

DCSAMP.C es un ejemplo de un programa que usa llamadas al dispositivo

BCSAMP.C es un ejemplo de un programa que usa llamadas a la tarjeta.
 CSAMP488.C es un ejemplo de un programa que usa funciones 488.2 .

Para poder crear programas de aplicación en C es necesario incluir la siguiente directiva al comienzo de cada programa:

#include "decl.h"

El archivo MCIB.OBJ es la interfaz entre el lenguaje C para la NI-488.2 y para el programa en ambiente MS-DOS. Entonces el programa de aplicación escrito y compilado tiene que ser encadenado con el archivo objeto MCIB.OBJ para producir un archivo ejecutable.

Antes de pasar al ejemplo de programación, se tiene que mencionar las siguientes operaciones que dan facilidades para programar en todas las opciones de lenguajes de programación.

- Palabra de estado (ibsta).
- Códigos de error (iberr).
- Variables de conteo (ibcut, ibcntl).
- Terminación de lectura y escritura.

Estas operaciones nos ponen al día sobre el estado del dispositivo o de la tarjeta a la que se desee.

Palabra de estado (ibsta). Todas las funciones y rutinas regresan una palabra de estado que contiene información del estado de la GPIB. Sirve para probar condiciones y poder tomar la decisión de si se quiere continuar con el proceso. La palabra de estado es regresada en la variable *ibsta*. Esta palabra contiene 16 bits. Un bit con valor de (1) indica que la condición es efectiva, un valor de (0) indica que la condición no es efectiva.

La tabla 3 indica la lista de condiciones su bit de posición, si es para dispositivos (disp), o para tarjetas GPIB (tarj), su descripción, y su mnemónico.

Mnemónicos.	Bit de posl.	Tipo de función.	Descripción.
ERR	15	disp,tarj	Error GPIB.
TIMO	14	disp,tarj	Excede el límite de tiempo.
END	13	disp,tarj	Detecta END o EOS.
SRQI	12	Tarj	Recibe una interrupción SRQ.
RQS	11	disp	Dispositivo que requiere servicio.
EVENT	10	tarj	Menciona que un evento a ocurrido
SPOLL	9	tarj	La tarjeta a sido encuestada serialmente por el controlador
CMPL	8	disp,tarj	I/O completa
LOK	7	tarj	Estado de bloqueo
REM	6	tarj	Estado remoto
CIC	5	tarj	Controlador en carga
ATN	4	tarj	Atención
TACS	3	tarj	Locutor
LACS	2	tarj	Oyente
DTAS	1	tarj	Estado de disparo en el dispositivo
DCAS	0	tarj	Limpia el dispositivo

Tabla 3 Indica las señales que contiene la palabra de estado (ibsta).

Códigos de error (iberr). Si el bit de ERR esta activo en la palabra de control, Significa que un error en la GPIB ha ocurrido. entonces se regresa un código de error en la variable **iberr** . La tabla 4 indica una lista con los códigos de error.

Mnemónico	Valor decimal	Explicación.
EDVR	0	DOS error
ECIC	1	Función requerida por la GPIB para ser CIC
ENOL	2	Escribe un error de manipulación de datos (handshake)
EADR	3	Tarjeta GPIB no direccionada correctamente
EARG	4	Argumento no valido para llamada de función
ESAC	5	La gpib no requiere de un sistema controlador
EABO	6	I/O operación abortada
ENEB	7	No existe tarjeta GPIB
EDMA	8	Para Windows 3 únicamente
EOP	10	I/O asíncrona en progreso
ECAP	11	Incapacidad de operación
EFSO	12	Error de sistema de archivo
EBUS	14	Error en el bus GPIB
ESTB	15	Sobreflujo en el byte de estado de encuesta serie
ESRQ	16	SRQ clavado en la posición ON
ETAB	20	Problema de escritura

Tabla 4. Códigos de error.

Variables de conteo (ibcnt, ibcntl). Las variables de conteo son actualizadas cada lectura, escritura o comando de función, con el numero de bytes transferidos por la operación actual. **ibcnt** es un valor entero de 16 bits, **ibcntl** Es un valor entero de 32 bits.

Terminación de lectura y escritura. Las especificaciones IEEE-488 definen dos métodos de identificación del ultimo byte de mensajes (datos) dependientes de un dispositivo. Estos métodos son incluir mensajes de fin por el locutor y pueden ser EOI y el EOS.

Programación por rutinas NI-488.2.

Aqui se hace un resumen de las diferentes rutinas de programación empleadas para realizar programas de aplicación en Microsoft C, en el manual de esta tarjeta se incluyen ejemplos de aplicación para cada rutina por lo que si se esta interesado se puede consultar el manual para un mayor entendimiento.

Sintaxis.	Descripción.
AllSpoll (tarjeta, dirección, resultado)	Encuesta en serie a todos los dispositivos.
DevClear (tarjeta, dirección)	Limpia un dispositivo.
DevClearList (tarjeta, dirección)	Limpia a múltiples dispositivos.
EnableLocal (tarjeta, dirección)	Activa las operaciones del panel frontal del dispositivo
EnableRemote (tarjeta, dirección)	Activa la programación de dispositivos en forma remota con la GPIB

FindLstn (Tarjeta, dirección, resultado, límite)	Busca a todos los dispositivos que son escuchas hasta un máximo especificado en límite.
FindRQS (tarjeta, dirección, resultado)	Determina cual dispositivo requiere el servicio.
GenerateREQF (tarjeta, dirección)	Cancela la petición del servicio.
GenerateREQT (tarjeta, dirección)	Petición de servicio.
GotoMultAddr (tarjeta, tipo, dirección, encuestaserie)	Activa y soporte para múltiples direcciones primarias o secundarias.
PasControl (tarjeta, dirección)	Transfiere el control a otro dispositivo, con capacidad de ser controlador.
PPoll (tarjeta, resultado)	Ejecuta una encuesta paralelo.
PPollConfig (tarjeta, dirección, linadatos, sentido)	Configura un dispositivo para encuestas paralelo.
PPollUnconfig (tarjeta, dirección)	Desconfigura dispositivos para encuestas paralelo.
RcvRespMsg (tarjeta, dato, terminación)	Lee bytes de datos de la dirección del dispositivo.
ReadStatusByte (tarjeta, dirección, resultado)	Realiza una encuesta serial a un dispositivo para conocer su byte de estado.
Receive (tarjeta, dirección, contador, terminación)	Lee bytes de datos de un dispositivo GPIB.
ReceiveSetup (tarjeta, dirección)	Prepara a un dispositivo en particular para enviar bytes de datos, y prepara a la GPIB para leer a estos.
ResetSys (tarjeta, dirección)	Inicializa un sistema GPIB
Send (tarjeta, dirección, dato, bandera)	Envía bytes de datos a un solo dispositivo GPIB.
SendCmds (tarjeta, comandos, contador)	Envía comandos GPIB.
SendDataBytes (tarjeta, dato, contador, bandera)	Envía bytes de datos a la dirección del dispositivo.
SendIfc (tarjeta)	Limpia la interfaz GPIB.
SendList (tarjeta, dirección, dato, contador, bandera)	Envía bytes de datos a múltiples dispositivos GPIB.
SendLlo (tarjeta)	Envía un mensaje de bloque local a todos los dispositivos.
SendSetUp (tarjeta, dirección)	Prepara un dispositivo particular para recibir bytes de datos.
SetRWLS (tarjeta, dirección)	Pone un dispositivo en particular en modo remoto, con estado de bloqueo.
TestSRQ (tarjeta, resultado)	Determina el estado actual de la línea SRQ.
TestSys (tarjeta, dirección, resultado)	Realiza una autoprueba
Trgger (tarjeta, dirección)	Dispara un dispositivo.
Triggerlist (tarjeta, dirección)	Dispara múltiples dispositivos.
WaitSRQ (tarjeta, resultado)	Espera asta que un dispositivo confirma una petición de servicio.

Pogramación por funciones NI-488.2.

Aqui se hace un resumen de las diferentes funciones de programación empleadas para realizar programas de aplicación en Microsoft C en el manual de esta tarjeta se incluyen ejemplos de aplicación para cada función por lo que si se esta interesado se puede consultar el manual para un mayor entendimiento.

Sintaxis.	Descripción.
ibbna (dispositivo, acceso)	Cambia al dispositivo de tarjeta
ibbac (dispositivo, v)	Toma el control activo
ibcl (dispositivo)	Limpia al dispositivo indicado.
ibcmd (dispositivo, comandos, contador)	Manda comandos en forma de cadenas.
ibcmda (dispositivo, comandos, contador)	Manda comandos de manera asincrónica y en forma de cadena
ibconfig (dispositivo, opción, valor)	Configura el software
dispositivo= ibdev (índice, pad, sad, tmo, eot, eos)	Abre e inicializa un dispositivo cuyo nombre se desconoce.
ibdma (dispositivo, v)	Activa/desactiva DMA.
ibeos (dispositivo, v)	Cambia el modo EOS
ibeot (dispositivo, v)	Activa/desactiva el mensaje END.
ibevent (dispositivo, evento)	Regresa el próximo evento
dispositivo= ibfind (unidad)	Abre un dispositivo y regresa la descripción de la unidad
ibgts (dispositivo, v)	Va de un estado controlador activo a un estado de espera.
ibist (dispositivo, v)	Pone o quita el bit de estado de la encuesta paralela.
iblines (tarjeta, líneas)	Obtiene el estado de las líneas GPIB.
ibln (pad, sad, locutor)	Checa la presencia del dispositivo en el bus.
ibloc (dispositivo)	Pone en local al dispositivo.
ibonl (dispositivo, v)	Pone al dispositivo o tarjeta en on/off
ibpad (dispositivo, v)	Cambia la dirección primaria.
ibpct (dispositivo)	Transfiere el control
ibppc (dispositivo, v)	Configura la encuesta paralela
ibrd (dispositivo, lectura, contador)	Lec una cadena de datos.
ibrda (dispositivo, lectura, contador)	Lec en forma asincrónica una cadena de datos.
ibrdf (dispositivo, archivo)	Lec datos de un archivo.
ibrpp (dispositivo, ppr)	Conduce una encuesta paralela
ibrsc (dispositivo, v)	Regresa el byte de la encuesta serial.
ibrsv (dispositivo, v)	Se requiere un servicio, cambia o activa la encuesta serial.
ibsad (dispositivo, v)	Cambia o desactiva la dirección secundaria.
ibsic (dispositivo)	Manda la interfaz limpia por 100 μ seg
ibsr (dispositivo, v)	Activa o desactiva la línea de manejo remoto
ibsrq (función)	Registra una rutina de interrupción SRQ
ibstop (dispositivo)	Aborta una operación asincrónica.
ibtmo (dispositivo, v)	Cambia o desactiva el tiempo límite.
ibtrp (máscara, modo)	Configura las aplicaciones monitor
ibtrg (dispositivo)	Dispara al dispositivo.
ibwait (dispositivo, máscara)	Espera para selección de evento.
ibwrt (dispositivo, escritura, contador)	Escribe un dato en forma de cadena
ibwrt (dispositivo, escritura, contador)	Escribe un dato asincrónico de una cadena.
ibwrtf (dispositivo, archivo)	Escribe datos de un archivo

En este proyecto y solo para fines didácticos se programaron algunas secuencias de control para el analizador 2133 de B&K, en las que se definía el "sistema de arranque ó setup" para adquirir una serie de datos de intensidad. En el proyecto hay que recordar que se utiliza el programa WT9378 de B&K, quien se encarga de programar al analizar de frecuencias y habilitarlo para capturar información de intensidad sonora, por lo que ya no fue necesario hacer un programa propio que hiciera lo anterior, obviamente es

necesario la utilización de la tarjeta con interfaz IEEE-488, lo que implica su programación la cual se explico anteriormente.

Para finalizar es conveniente consultar el manual de la GPIB, para complementar la información anterior y conocer en detalle la forma de utilizar esta interfaz.

APENDICE D

Instrucciones para el manejo del programa WT9378 Después de instalar el programa, no olvidar colocar la llave de protección en el puerto paralelo, para que funcione.

El programa cuando esta corriendo muestra una barra de menú, con cinco opciones:

Config para personalizar el programa y especificar los elementos que componen al sistema de medición.

Input selecciona archivos de datos, transfiere datos al disco, selecciona la forma en que se desea tomar los datos.

Mapping para desplegar la representación gráfica de la intensidad sonora

Power para mostrar la distribución de potencia acústica sobre la superficie de medición.

Quit para terminar el programa.

La opción *Config* tiene a su vez tres opciones:

- *Edit*
- *Store*
- *Default*

Seleccionando la opción *Edit*, se puede cambiar la configuración del programa, es decir que se puede editar el programa o bien manejar parámetros ya establecidos. Mediante esta opción se pueden especificar dos modos de trabajo para la adquisición de datos, una es modo normal (ó manual ya que la transferencia de datos es manual usando el teclado de la computadora o las teclas del analizador) y el otro en modo robot, este modo requiere de un sistema que mueva a la sonda a los puntos apropiados en la superficie de medición, La sonda toma las lecturas las cuales son automáticamente transferidas al analizador, obviamente para el proyecto se puso esta opción en modo robot. La opción *Store*, es para guardar los cambios de la configuración en el disco, la opción *Default* configura el programa con parámetros ya establecidos.

La opción *Input* muestra un menú con cuatro opciones:

- *Catalog*
- *Analyzer*
- *Multispectrum*
- *Multichannel*

La opción *Catalog*, muestra una lista de los archivos de datos, con esta opción se pueden recuperar los datos de un archivo, o bien fusionar archivos.

Las opciones *Analyzer* y *Multichannel*, la selección de estas opciones depende del tipo de analizador y de las mediciones que se realizaran. Si se esta utilizando un analizador multicanal, se tiene que especificar primero en *Config*, para multichannel

también se tiene que especificar, aun cuando un analizador multicanal esta conectado, multicanal se refiere a la manera en que el analizador puede trabajar.

La opción *Multispectrum*, es para procesar y manipular una cadena o arreglo de datos como una sola unidad. Tanto *Multispectrum* como *Multichannel*, no son empleadas en este proyecto, por lo que solo se utiliza la opción *Analyzer*. Obviamente si se quiere saber y profundizar más, se aconseja consultar la documentación técnica de este programa.

Dentro de la opción *Analyzer*, es necesaria la selección del tipo de superficie que contiene los puntos de medición, hay tres tipos de superficie de medición que se pueden seleccionar y son:

* *Grid*

* *Hemisphere*

* *Any Area*

Las opciones *Hemisphere* y *Any Area*, definen, en primer termino, una superficie hemisférica que envuelva a la fuente sonora, esta superficie es la necesaria para el mapeo de potencia sonora, su geometría es como se muestra en la figura 2.2.1, la otra opción especifica una área con 12 sectores de medición máximo.

En el proyecto se selecciona la opción *Grid* ó reja, esta se trata de una reja, que no es más que un plano de superficie rectangular con puntos de medición en un sistema de coordenadas dados por columna y renglón. Nuestro robot permite un máximo de 16 x 12, puntos de medición, y el programa un máximo de 75 x 12. En este sentido el robot no cubre los 75 puntos sobre el eje *X* únicamente 16. Para fines prácticos y en concreto para la medición de intensidad sonora de la guitarra se puede cubrir con 16 x 12 una área suficiente como para obtener el mapa.

Cuando se ha seleccionado la superficie de medición (en este proyecto es la opción *Grid*), aparece en la pantalla una ventana, que muestra el analizador usado (2133), ademas se especifica la secuencia de adquisición de datos, es decir que tipo de movimiento tendrá que efectuar el robot para mover la sonda, para mejor entendimiento hay que recordar la figura 3.2.

También se pueden colocar líneas de texto para encabezado de nuestros mapas. Adicionalmente se tiene que definir el nombre del archivo en que se guardaran los datos. Dentro de esta misma ventana se tiene que especificar que tipo de datos se leerán (presión ó intensidad), y la dirección perpendicular respecto del plano, en este caso, son datos de intensidad en la dirección *z*. También hay que definir el ancho de banda a 1/3 de octava (para este proyecto), la ponderación se pone en lineal, hay que definir también el rango de frecuencia dependiendo del tamaño del espaciador que lleva la sonda, es decir que para un rango de 63 Hz a 5000 Hz se utiliza el espaciador de 12 mm, esta ventana permite la especificación de esto. Dentro de la misma ventana también hay que especificar la presión atmosférica en mBar, la temperatura en grados celsius, el tiempo de promediación en segundos, y la superficie a cubrir por el barrido en [m²].

Una vez que se ha configurado el programa, este se encarga de manera automática del control de todo el sistema y de la adquisición de los datos de intensidad y considerando que ya se completo el barrido, teniéndose los datos ya transferidos en un archivo, se pasa al análisis de esta información, que entrega los datos ya calculados para el mapeo.

Mapping es la opción que se refiere a las cuatro representaciones gráficas de intensidad sonora, que pueden ser.

- Mapa numérico: Esta opción despliega el mapa numérico de la intensidad sobre un dibujo correspondiente a la rejilla.
- Mapa de vectores: Es una representación vectorial de la intensidad sobre la rejilla.
- Mapa de contorno o de nivel. Esta opción da una representación gráfica sobre la rejilla uniendo por medio de interpolaciones los puntos con un nivel de intensidad igual.
- Mapa en 3-D: Es una representación visual en tres dimensiones de los niveles de intensidad sonora.

La interpolación se puede usar con los mapas de contorno y los de 3-D, solo se necesita dar el número de pasos entre las mediciones y el tipo de interpolación que se realizara, puede ser interpolación lineal o logarítmica.

La opción **Power** es para mostrar los parámetros analizados, a partir del espectro de potencia para diferentes bandas de frecuencia, esta opción entre otras cosas genera un archivo en modo de texto, el cual contiene una tabla con la siguiente información:

<i>Renglón</i> <i>n</i>	<i>Columna</i> <i>a</i>	<i>Intensidad</i> <i>ad</i> [dB]	<i>Área</i> [m ²]	<i>Potencia</i> <i>a</i> [dB]	<i>Dirección</i> <i>ón</i>	<i>Estado</i>
----------------------------	----------------------------	--	----------------------------------	-------------------------------------	-------------------------------	---------------

La información que contiene este archivo fue utilizada para el procesamiento posterior con un programa hecho en Matlab, ya que entrega el nivel y la dirección de la intensidad sonora en cada punto de medición.

Es recomendable consultar el manual de este programa, para profundizar en cuanto a su forma de configuración y de utilización

BIBLIOGRAFIA

1. E. Skudrzyk, *The foundations of acoustic*, (Ed. Springer – verlag, New York),(1971)
2. L E. Kinsler, A. R. Frey, A. B. Coppens, *Fundamentos de acústica* (Ed. Limusa, México), (1988)
3. F.J. Fahy, *Sound Intensity* (Ed. Elsevier, New York), (1989)
4. Brüel & Kjær, *Sound Intensity*, Technical Documentation.
5. R. Upton, *Sound intensity-A powerful new measurement tool*, Sound and Vibration Oct, 10-18, (1982)
6. M. J. Crocker, F. Jacobsen, *Sound intensity*, Encyclopedia of acoustics (Ed. John Wiley & Sons), (1997)
7. F. Cervera, F. Belmar, H. Etelles, *A low cost robot for acoustic intensity measurements*, Noise Control Eng. J. 42(1), 31-35, (1994)
8. K. Takashi, A. Sugawara, *Stepping motors and their microprocessors controls*, (Ed. Clarendon), (1994).
9. Crydom, *Solid Stated Relay Handbook*(1990)
10. PC-LabCards, *User's Manual Digital I/O & Counter Card Model PCL-720*,(1988)
11. A. Birkett, *Electronics World Jun*, 502, 1997
12. PC-LabCard, *PCLD-782 16 Channel Opto-isolated D/I Board*
13. Microsoft Corporation, *Run-Time Library Reference Microsoft C/C++ Version 7.0*
14. B. W. Kernighan, D. M. Ritchie, *El lenguaje de programación C*, (Prentice-Hall) (1987)
15. D.V. Gadre, K. Pramod, V. S. Varma, *Computers in Physics*, vol. 8, (1989)
16. J M. Poblet, *Interconexión de periféricos a microprocesadores*, (Ed. Marcombo), (1984)
17. National Instruments, *NI-488.2 Software Reference Manual for MS-DOS*, (1993).
18. Brüel & Kjær, *Instruction Manual Analyzer type 2133*, Technical Documentation.
19. Brüel & Kjær, *Mapping and Sound Power Program WT9378*, Technical Documentation, (1993).
20. Brüel & Kjær, *Sound intensity calibrator type 3541*, Technical Documentation.
21. A. Caldersmith, *The guitar frequency response*, Journal of Guitar Acoustic, Sept. (1982)
22. E.V. Jansson, *A study of acoustical and hologram interferometric measurements of the top plate vibrations of a guitar*, Acustica. Vol. 25 (1971)
23. Brüel & Kjær, *Sound intensity calibrator type 3541*, Technical Documentation.
24. A. Caldersmith, *The guitar frequency response*, Journal of Guitar Acoustic, Sept. (1982)
25. E.V. Jansson, *A study of acoustical and hologram interferometric measurements of the top plate vibrations of a guitar*, Acustica. Vol. 25 (1971)