ESCUELA POLITÉCNICA NACIONAL

ESCUELA DE INGENIERÍA

DISEÑO Y CONSTRUCCIÓN DE UN AMPLIFICADOR DE POTENCIA PARA AUDIO CONTROLADO POR COMPUTADORA

PROYECTO PREVIO A LA OBTENCIÓN DEL TÍTULO DE INGENIERO EN ELECTRÓNICA Y TELECOMUNICACIONES

ÁLVAREZ CASTILLO VÍCTOR HUGO
ARROBO VIDAL GABRIEL EDUARDO

DIRECTOR: ING. MIGUEL HINOJOSA

Quito, Noviembre de 2003

DECLARACIÓN

Nosotros, VÍCTOR HUGO ÁLVAREZ CASTILLO y GABRIEL EDUARDO ARROBO VIDAL, declaramos bajo juramento que el trabajo aquí descrito es de nuestra autoría; que no ha sido previamente presentada para ningún grado o calificación profesional; y, que hemos consultado las referencias bibliográficas que se incluyen en este documento.

A través de la presente declaración cedemos nuestros derechos de propiedad intelectual correspondientes a este trabajo, a la Escuela Politécnica Nacional, según lo establecido por la Ley de Propiedad Intelectual, por su Reglamento y por la normatividad institucional vigente.

Víctor Álvarez

Gabriel Arrobo

CERTIFICACIÓN

Certifico que el presente trabajo fue desarrollado por Víctor Hugo Álvarez Castillo y Gabriel Eduardo Arrobo Vidal, bajo mi supervisión.

Msc. Miguel Hinojosa

DIRECTOR DE PROYECTO

AGRADECIMIENTO

A *nuestro* director de proyecto Msc. Miguel Hinojosa, por su apoyo y dirección acertada, para la culminación exitosa del presente proyecto.

A todas las personas que de alguna manera colaboraron en nuestra formación y realización del presente trabajo.

Un agradecimiento especial a la Escuela Politécnica Nacional que nos brindó la oportunidad de superarnos y poder salir adelante.

DEDICATORIA

A Dios y a mis padres,
quienes con su amor y esfuerzo
me han permitido dar un gran paso en mi vida.

A mis hermanos y familia
por su apoyo incondicional
que día a día me brindan.

Álvarez Castillo Víctor Hugo

DEDICATORIA

A Dios, mis padres,
mis hermanos, mi tía Piedad
mi familia, mis amigos
que siempre están brindándome
apoyo en todo momento
para seguir adelante

Gabriel Eduardo

RESUMEN

El presente trabajo está enfocado para servir como una guía, en la realización del control de dispositivos mediante computador. En este caso el dispositivo a controlar es un sistema de amplificación de audio.

El fin de este proyecto es mostrar la facilidad, versatilidad, efectividad entre otras ventajas que da el control realizado mediante computador.

En el primer capítulo se presenta las etapas de un amplificador de audio, acompañado de las estructuras electrónicas que comúnmente conforman estas etapas. Con estas bases se realiza una breve explicación de la estructura electrónica interna de un circuito integrado amplificador de audio de 25 vatios (STK 4141), para reforzar la comprensión del tema se realiza un diseño de un amplificador de 25 vatios sobre la base de elementos discretos.

En el segundo capítulo se detalla el hardware necesario para la adquisición y envío de datos, se da una visión global del funcionamiento de las tarjetas de adquisición de datos, los elementos con los que cuenta, así como también ciertas características.

En el tercer capítulo se enseña como realizar la programación adecuada para que junto con el hardware de adquisición de datos poder realizar las respectivas adquisiciones o envíos de información desde y hacia diferentes sistemas para ser controlados vía computadora.

En el cuarto capítulo se da una visión global del sistema, es decir, el funcionamiento entre el amplificador de potencia de audio, el hardware de adquisición de datos, la interfaz de usuario y el usuario.

En el quinto capítulo se dan a conocer las conclusiones a las que se ha llegado después de haber realizado el proyecto, así como también las respectivas

recomendaciones para que en la práctica sirvan de ayuda en otras investigaciones y/o desarrollos.

Al final se presenta algunos documentos adicionales en los cuales se puede encontrar información adicional con mayor profundidad, si es el caso.

PRESENTACION

En la actualidad, el desarrollo de la tecnología ha incentivado a la humanidad a buscar herramientas que le ayuden a realizar su trabajo más efectiva y eficientemente.

Para el buen manejo de ciertos equipos, de tal manera que se obtengan buenos resultados se necesita que se cumpla algunas condiciones, por ejemplo: precisión, exactitud, rapidez de ejecución, requerimiento de parámetros, etc.; acciones que los humanos estamos limitados para realizar con efectividad.

Por ésta necesidad se ha desarrollado el manejo o control de ciertos equipos mediante computador, aprovechando las capacidades que tiene el computador, como son: su velocidad de procesamiento de información, su capacidad para almacenar información, la precisión y exactitud que pueden tener sus resultados, entre otras.

El control por computador en si es la forma como se adquiere la información del equipo, el procesamiento de esta información y el regreso de la información procesada al equipo para que este realice una determinada tarea.

Para que un equipo pueda ser controlado por computador, necesita que se le adicionen ciertas modificaciones, ya sea internas o externas para que se pueda comunicar con el computador, lo que se conoce como la interfaz hardware. El manejo de este interfaz se lo realiza mediante un software residente en el computador que también se lo conoce como interfaz software.

El presente proyecto persigue guiar a otras personas en el control de sistemas o equipos mediante computador.

ÍNDICE

CAPÍTULO 1: AMPLIFICADORES DE POTENCIA DE AUDIO 1

1.1 INTRODUCCIÓN 2

1.2 AMPLIFICADORES DE POTENCIA 4

1.2.1 DIAGRAMA DE BLOQUES *5*

1.2.1.1 Etapa De Entrada 5

1.2.1.2 Etapa Excitadora 6

1.2.1.3 Etapa De Potencia o De Salida 6

1.2.1.4 Realimentación 7

1.2.1.5 Fuente De Alimentación 15

1.3 CLASES DE AMPLIFICADORES DE POTENCIA 18

1.3.1 CLASE A *18*

1.3.2 CLASE B *19*

1.3.3 CLASE AB *21*

1.3.4 CLASE C *21*

1.3.5 CLASE D *22*

1.3.6 CLASE E *23*

1.3.7 CLASE G *24*

1.3.8 CLASE H *24*

1.4 PARÁMETROS CARACTERÍSTICOS DE LOS AMPLIFICADORES DE POTENCIA 24

1.4.1 POTENCIA ENTREGADA POR LA FUENTE DE ENERGÍA *24*

1.4.2 POTENCIA ÚTIL ENTREGADA A LA CARGA *25*

1.4.3 RENDIMIENTO DEL AMPLIFICADOR *26*

1.4.4 DISTORSIÓN *26*

1.4.4.1 Distorsión armónica 26

1.4.4.1.1 Distorsión Armónica Total (THD) 27

1.5 AMPLIFICADORES DE AUDIO 27

1.5.1 AMPLIFICADOR DE AUDIO EN CONTRAFASE CUASI-COMPLEMENTARIO *28*

1.5.1.1 Par Darlington 29

1.5.1.2 Par Retroalimentado 32
1.6 CIRCUITOS INTEGRADOS 37
1.6.1 ESTRUCTURA INTERNA DE UN CIRCUITO INTEGRADO AMPLIFICADOR DE AUDIO *38*
1.6.1.1 Fuentes de Corriente 39
1.6.1.2 Espejos de Corriente 41
1.6.1.3 Amplificadores Diferenciales 42
1.6.1.4 STK 4141 46
1.7 MODOS DE CONEXIÓN A LA SALIDA 54
MODO ESTÉREO *54*
1.7.1 MODO PARALELO O MONO *55*
1.8 CARACTERÍSTICAS PRINCIPALES DE LOS AMPLIFICADORES COMERCIALES 56
1.8.1 POTENCIA DE SALIDA *56*
1.8.1.1 Potencia Nominal 56
1.8.1.2 Potencia Musical o Dinámica 56
1.8.2 RESPUESTA EN FRECUENCIA. *57*
1.8.3 SLEW RATE *58*
1.8.4 SENSIBILIDAD DE ENTRADA *59*
1.8.5 IMPEDANCIA DE ENTRADA *60*
1.8.6 IMPEDANCIA DE SALIDA *60*
1.8.7 FACTOR DE AMORTIGUAMIENTO (FA) *61*
1.8.8 RENDIMIENTO *62*
1.9 PREAMPLIFICADORES Y MEZCLADORES 63
1.9.1 PREAMPLIFICADORES *63*
1.9.1.1 Amplificadores de Voltaje 64
1.9.1.1.1 Amplificador No Inversor 64
1.9.1.1.2 Amplificador Inversor 66
1.9.2 MEZCLADORES *69*
1.9.2.1 Mezclador No Inversor 69
1.9.2.2 Mezclador Inversor 70
1.10 PROTECCIONES DE DISPOSITIVOS Y DE CIRCUITOS 70
1.10.1 ENFRIADORES Y DISIPADORES DE CALOR *71*

1.10.2 TRANSITORIOS DEL LADO DE LA ALIMENTACIÓN Y DEL LADO DE LA CARGA *72*
1.10.3 PROTECCIONES CONTRA SOBRE CORRIENTE *74*
1.10.4 PROTECCIONES CONTRA TENSIÓN CONTINUA *75*
1.11 DISEÑO DEL SISTEMA DE AUDIO 75
1.11.1 DISEÑO DE LOS PREAMPLIFICADORES *76*
1.11.1.1 Requerimientos y especificaciones 76
1.11.1.2 Diseño de Preamplificadores para Micrófonos 77
1.11.1.3 Diseño de Preamplificadores para Entradas Auxiliares 80
1.11.2 DISEÑO DEL AMPLIFICADOR DE POTENCIA *81*
1.11.2.1 Requerimientos y especificaciones 81
1.11.2.2 Diseño del amplificador: 82
1.11.3 DISEÑO DEL MEZCLADOR *101*
1.12 MONTAJE E IMPLEMENTACION 101
1.12.1 Fuente de Alimentación *101*
1.12.2 Etapa de preamplificación *102*
1.12.2.1 Características Principales: 103
1.12.2.2 Estructura Física: 104
1.12.2.3 Estructura Interna: 104
1.12.3 Etapa de Potencia *105*
1.13 PRUEBAS Y RESULTADOS 106
1.13.1 Fuente de alimentación *106*
1.13.2 Etapa de preamplificación *106*
1.13.3 Etapa de Amplificación *107*
CAPÍTULO 2: INTERFAZ HARDWARE **108**
2.1 INTRODUCCIÓN 109
2.2 DEFINICIÓN 109
2.3 DESCRIPCIÓN 110
2.3.1 Proceso de Adquisición de Datos *111*
2.3.1.1 Sensor / Actuador (Transductores) 111
2.3.1.2 Acondicionamiento de Señal 111
2.3.1.2.1 Amplificación 112

2.3.1.2.2 Filtrado 112
2.3.1.2.3 Aislamiento 112
2.3.1.2.4 Multiplexación 112
2.3.1.2.5 Fuentes de Excitación 113
2.3.1.3 Hardware de Adquisición de Datos 113
2.3.1.3.1 Entrada Analógica 114
2.3.1.3.2 Salida Analógica 114
2.3.1.3.3 Entrada / Salida Digital 114
2.3.1.3.4 Contador / Temporizador 114
2.3.1.4 Computador 115
2.3.1.5 Software 115
2.3.1.5.1 Software Controlador 116
2.3.1.5.2 Software de Aplicación 116
2.3.2 Descripción del sistema a implementarse *117*
2.4 ESPECIFICACIÓN DE REQUERIMIENTOS 118
2.5 DISEÑO 118
2.6 ELEMENTOS UTILIZADOS 120
2.6.1 Tarjeta de adquisición de datos *120*
2.6.1.1 Características 124
2.6.1.1.1 Entradas analógicas 124
2.6.1.1.2 Salidas analógicas 134
2.6.1.1.3 Entradas y salidas digitales 135
2.6.1.1.4 Los circuitos de temporización y contadores de entrada y salida 135
2.6.1.2 EJEMPLO DE TARJETA DE ADQUISICIÓN DE DATOS 136
2.6.1.2.1 Características Generales 137
2.6.1.2.2 Configuración 138
2.6.1.2.3 Diagrama de bloques 139
2.6.1.2.4 Interfaz 82C55 139
2.6.1.2.5 Interfaz 82C54 142
2.6.1.2.6 Conversor Analógico – Digital ADS 7800 143
2.6.2 PotenciómetroS *145*
2.6.2.1 Símbolo: 145
2.6.2.2 Potenciómetros Digitales 146

2.6.2.2.1 Características Generales 146
2.7 MONTAJE, PRUEBAS Y AJUSTES 149
2.8 FUNCIONAMIENTO 150
CAPÍTULO 3: INTERFAZ SOFTWARE 151
3.1 INTRODUCCIÓN 152
3.2 DEFINICIÓN 153
3.3 DESCRIPCIÓN 154
3.3.1 Herramienta de Adquisición de datos 154
3.3.1.1 Funciones 155
3.3.1.2 Motor de Adquisición de datos 156
3.3.1.3 Adaptador para el manejo de Hardware 156
3.3.1.4 Información de la herramienta de Adquisición de datos 156
3.3.2 Software de configuración y pruebas de la tarjeta de adquisición de datos 158
3.3.3 Comenzar a trabajar con las tarjetas de Adquisición de datos 164
3.3.3.1 Adquisición de Datos 164
3.3.3.2 Envío de Datos 165
3.3.3.3 Lectura / Escritura de valores Digitales 166
3.3.3.4 Ejemplo de la creación de un objeto 168
3.3.4 Creación de Interfaz de usuario 171
3.3.4.1 Creación de forma gráfica 172
3.3.4.1.1 Controles para interfaz de usuario 172
3.3.4.1.2 Propiedades de los controles 178
3.3.4.2 Creación por código 180
3.4 ESPECIFICACIÓN DE REQUERIMIENTOS 182
3.5 DISEÑO 183
3.6 IMPLEMENTACIÓN 184
3.6.1 Creación de las entradas y salidas analógicas 191
3.6.2 Creación de las entradas / salidas digitales 194
3.6.3 Creación de la pantalla de usuario 195
3.7 PRUEBAS 197
3.8 UTILIZACIÓN 198

CAPÍTULO 4: PRUEBAS EN CONJUNTO 201

4.1 INTRODUCCIÓN 202

4.2 PARTES DEL EQUIPO 202

4.3 FUNCIONAMIENTO Y MODO DE OPERACION 203

4.4 PRUEBAS 204

4.5 RESULTADOS 205

4.6 EQUIPOS DE MEDICIÓN 206

4.7 VISIÓN GLOBAL DE SISTEMA 207

4.8 COSTO DEL PROTOTIPO 208

CAPÍTULO 5: CONCLUSIONES Y RECOMENDACIONES 211

5.1 CONCLUSIONES 212

5.2 RECOMENDACIONES 215

ANEXO A: ESQUEMÁTICOS Y RUTEO DE LA PLACAS A.1

ANEXO B: CÓDIGO DE LOS PROGRAMAS B.1

ANEXO C: HOJAS DE DATOS C.1

INDICE DE FIGURAS

FIGURA 1.1) Partes de un Sistema de Audio 3
FIGURA 1.2) Proceso de amplificación de una señal 4
FIGURA 1.3) Diagrama de bloques de una amplificador de potencia 5
FIGURA 1.4) Diagrama de bloques de un circuito con Retroalimentación Negativa 8
FIGURA 1.5) Diagrama de bloques de los tipos de realimentación negativa 10
FIGURA 1.6) Diagrama de bloques de la entrada de un circuito con realimentación RSDP 12
FIGURA 1.7) Diagrama de bloques de la salida de un circuito con retroalimentación RSDP 14
FIGURA 1.8) Diagrama de una Fuente de Alimentación DC variable 16
FIGURA 1.9) Ciclo de operación de un amplificador Clase A 19
FIGURA 1.10) Circuito en contrafase de simetría complementaria 20
FIGURA 1.11) Distorsión en el cruce por cero en un amplificador clase B 20
FIGURA 1.12) Circuito de potencia Clase C 22
FIGURA 1.13) Diagrama explicativo del funcionamiento de un Amplificador Clase D 23
FIGURA 1.14) Amplificador de audio cuasi-complementario 28
FIGURA 1.15) Circuito par Darlington 29
FIGURA 1.16) Circuito amplificador con par Darlington 30
FIGURA 1.17) Circuito AC equivalente de la figura 1.16 30
FIGURA 1.18) Circuito AC equivalente con un voltaje V_x a la salida y cortocircuitando a la fuente voltaje de la figura 1.16 31
FIGURA 1.19) Circuito Par retroalimentado 32
FIGURA 1.20) Circuito amplificador con par retroalimentado 33
FIGURA 1.21) Circuito AC equivalente 1 del circuito de la figura 1.20 33
FIGURA 1.22) Circuito AC equivalente 2 del circuito de la figura 1.20 33
FIGURA 1.23) Circuito AC equivalente con un voltaje V_x a la salida y cortocircuitando a la fuente voltaje de la figura 1.20 35
FIGURA 1.24) Circuito fuente de corriente 39

FIGURA 1.25) Fuente de corriente con un diodo zener 40
FIGURA 1.26) Circuito espejo de corriente 41
FIGURA 1.27) Flujo de corrientes en un circuito espejo de corriente 42
FIGURA 1.28) Circuito amplificador diferencial 43
FIGURA 1.29) Circuito AC equivalente del circuito de la figura 1.28 44
FIGURA 1.30) Amplificador diferencial con fuente de corriente conectada a los emisores para aumentar la relación de rechazo en modo común. 46
FIGURA 1.31) Diagrama interno del CI STK4141 46
FIGURA 1.32) Etapas que conforman un amplificador de audio 47
FIGURA 1.33) Estructuras que conforman el STK4141 48
FIGURA 1.34) Estructura 6 vista de otra manera 52
FIGURA 1.35) Estructura 6 con un capacitor de desvío para evitar señales parásitas 53
FIGURA 1.36) Circuito de un amplificador de audio estereo de 25 Vatios basado en el C.I. STK 4141 53
FIGURA 1.37) Diagrama de bloques de un circuito en modo estereo 54
FIGURA 1.38) Diagrama de bloques de un circuito en modo paralelo o mono 55
FIGURA 1.39) Respuesta de frecuencia de un altavoz 57
FIGURA 1.40) Efecto de triangulación por un Slew rate pequeño 59
FIGURA 1.41) Amplificador de voltaje No inversor 65
FIGURA 1.42) Diagrama que muestra las impedancias en los diferentes puntos de un amplificador No inversor. 65
FIGURA 1.43) Circuito amplificador de voltaje Inversor 67
FIGURA 1.44) Circuito amplificador de voltaje Inversor 68
FIGURA 1.45) Circuito mezclador No Inversor 69
FIGURA 1.46) Circuito mezclador inversor 70
FIGURA 1.47) Circuito eléctrico equivalente de un sistema de disipación. 71
FIGURA 1.48) Capacitor y resistencia conectados en el secundario para eliminar voltaje transitorio 73
FIGURA 1.49) Circuito equivalente en el momento de la desconexión 73
FIGURA 1.50) Capacitor y resistencia conectados en paralelo a la carga para eliminar voltaje transitorio 74
FIGURA 1.51) Diagrama de bloques de un sistema de audio 76

FIGURA 1.52) Circuito preamplificador para una entrada de micrófono 78
FIGURA 1.53) Circuito de preamplificador para una entrada auxiliar 80
FIGURA 1.54) Circuito Amplificador de Audio a diseñarse 83
FIGURA 1.55) Circuito de protección contra cortocircuito o sobre tensiones en la carga 95
FIGURA 1.56) Circuito Amplificador de Audio diseñado (para un canal) 100
FIGURA 1.57) Circuito mezclador de cuatro entradas 101
FIGURA 1.58) Fuente de alimentación 102
FIGURA 1.59) Diagrama de pines del TL084 104
FIGURA 1.60) Estructura interna del TL084 104
FIGURA 2.1) Diagrama de bloques de un sistema de Adquisición de datos 111
FIGURA 2.2) Diagrama de bloques de los subsistemas 113
FIGURA 2.3) Relación Usuario – Software – Hardware 116
FIGURA 2.4) Diagrama de bloques del sistema a implementarse 117
FIGURA 2.5) Canal analógico simple (modo común) 124
FIGURA 2.6) Canal analógico diferencial (modo diferencial) 125
FIGURA 2.7) Onda que representa la variación de temperatura en un tanque 125
FIGURA 2.8) Onda que representa a una señal de audio 126
FIGURA 2.9) Proceso de conversión Digital-Análogo-Digital 126
FIGURA 2.10) Proceso de digitalización de una señal analógica 127
FIGURA 2.11) Señal analógica muestreada 129
FIGURA 2.12) Señales Unipolares y Bipolares de diferente amplitud 131
FIGURA 2.13) Proceso de Multiplexación de canales de entrada 132
FIGURA 2.14) Muestreo continuo 133
FIGURA 2.15) Muestreo simultáneo 133
FIGURA 2.16) Señal analógica obtenida del proceso de conversión digital-analógico 134
FIGURA 2.17) Tarjeta de adquisición de datos PCI-DAS1000 136
FIGURA 2.18) Cable para interconexión entre Tarjeta PCI y el conector terminal 136
FIGURA 2.19) Conector Terminal 137
FIGURA 2.20) Diagrama de bloques interno de la PCI-DAS1000 139
FIGURA 2.21) Diagrama de pines del interfaz 82C55 140

FIGURA 2.22) Diagrama de bloques interno del interfaz 82C55 141
FIGURA 2.23) Diagrama de pines del interfaz 82C54 142
FIGURA 2.24) Diagrama de pines del ADC ADS 7800 144
FIGURA 2.25) Diagrama de bloques interno del ADS 7800 144
FIGURA 2.26) Símbolo que representa a un potenciómetro 145
FIGURA 2.27) Diversos tipos de potenciómetros 145
FIGURA 2.28) Diagrama general de un potenciómetro digital Xicor X9Cxxx 147
FIGURA 2.29) Diagrama de bloques interno del potenciómetro digital Xicor X9Cxxx 148
FIGURA 3.1) Proceso del Software en la Adquisición de datos 155
FIGURA 3.2) Comando para obtener información de la herramienta de Adquisición de datos 156
FIGURA 3.3) Comando para obtener información cuales son las DAQ que se encuentran instaladas 157
FIGURA 3.4) Comando para obtener información de la DAQ de Computer Boards 157
FIGURA 3.5) Comando para obtener información de la Tarjeta de sonido 157
FIGURA 3.6) Comando para obtener información de los subsistemas con los que cuenta la DAQ "CBI" 158
FIGURA 3.7) Comando para obtener información de los subsistemas con los que cuenta la DAQ "Winsound" 158
FIGURA 3.8) Software *Instacal* para instalación y configuración de las DAQ de Computer Boards 159
FIGURA 3.9) Agregar una nueva DAQ de Computer Boards 160
FIGURA 3.10) Seleccionar la DAQ y el número de tarjeta que se va a agregar 160
FIGURA 3.11) Configuración de una DAQ de Computer Boards 161
FIGURA 3.12) Configuración de canales, fuente de contador, etc. 161
FIGURA 3.13) Calibración de el conversor Analógico / Digital 162
FIGURA 3.14) Ejecución de la calibración del ADC para los diferentes rangos de entrada 162
FIGURA 3.15) Pruebas con la DAQ: generación de señales y adquisición de las mismas 163

FIGURA 3.16) Pruebas con la DAQ: muestreo de diferentes señales de entradas 164
FIGURA 3.17) Matlab: Ejemplo de la creación de una entrada analógica 168
FIGURA 3.18) Matlab: Ejemplo de agregar canales a una entrada analógica 168
FIGURA 3.19) Matlab: Configurar la velocidad de muestreo y las muestras que se toman en cada canal de una entrada analógica 169
FIGURA 3.20) Matlab: Comenzar a adquirir datos con la entrada analógica 169
FIGURA 3.21) Matlab: Muestras adquiridas con la entrada analógica 169
FIGURA 3.22) Matlab: Gráfico de las muestras adquiridas con la entrada analógica 170
FIGURA 3.23) Matlab: Información de las propiedades de la entrada analógica 170
FIGURA 3.24) Matlab: Información de los voltaje de entrada máximos permitidos por la DAQ 171
FIGURA 3.25) Comunicación entre los subsistemas y los objetos de la herramienta de Adquisición de Datos 171
FIGURA 3.26) Matlab: Creación de interfaces de usuario con el comando "GUIDE" 172
FIGURA 3.27) Matlab: representación gráfica de Push Button 173
FIGURA 3.28) Matlab: representación gráfica de Toggle Button 173
FIGURA 3.29) Matlab: representación gráfica de Radio Button 174
FIGURA 3.30) Matlab: representación gráfica de CheckBox 174
FIGURA 3.31) Matlab: representación gráfica de Edit Text 175
FIGURA 3.32) Matlab: representación gráfica de Static Text 175
FIGURA 3.33) Matlab: representación gráfica de Slider 175
FIGURA 3.34) Matlab: representación gráfica de Frame, frame contiene varios controles mencionados anteriormente 176
FIGURA 3.35) Matlab: representación gráfica de List Box 176
FIGURA 3.36) Matlab: representación gráfica de Popup Menu 177
FIGURA 3.37) Matlab: representación gráfica de una señal aleatoria 177
FIGURA 3.38) Matlab: representación varios controles en la pantalla para crear interfaces de usuario 180
FIGURA 3.39) Matlab: programa de Inicio del controlador 184
FIGURA 3.40) Matlab: programa para configurar los subsistemas a utilizar 188

FIGURA 3.41) Matlab: Ejemplo de selección de subsistema de *Entradas Analógicas* 189
FIGURA 3.42) Matlab: Mensaje de error en la selección de subsistema de *Entradas Analógicas* 189
FIGURA 3.43) Matlab: Mensaje de error en la configuración 190
FIGURA 3.44) Matlab: Mensaje de advertencia por la no selección de los diferentes subsistemas 190
FIGURA 3.45) Matlab: programa Controlador del Amplificador de Potencia 195
FIGURA 3.46) Matlab: Gráfica del ecualizador 196
FIGURA 3.47) Matlab: FDATool herramienta para el diseño de filtros 196
FIGURA 3.48) Programa Matlab 199
FIGURA 4.1) Disposición interna de las etapas en el equipo amplificador de audio 203
FIGURA 4.2) Osciloscopio 206
FIGURA 4.3) Generador de señal 206
FIGURA 4.4) Multimetro 207
FIGURA 4.5) Estructura interna del equipo amplificador 207
FIGURA 4.6) Sistema global 208

INDICE DE TABLAS

TABLA 1.1) Rendimiento y ciclo de operación de las diferentes clases de amplificadores 18
TABLA 1.2) Modos de operación de amplificadores diferenciales 44
TABLA 1.3) Características del circuito de Muting 52
TABLA 1.4) Niveles de voltaje entregados por distintas fuentes 63
TABLA 1.5) Características de transistores 82
TABLA 1.6) Características de los transistores del amplificador diferencial 90
TABLA 1.7) Valores medidos de la fuente de energía 106
TABLA 1.8) Valores medidos en los preamplificadores 106
TABLA 1.9) Valores de Ganancia de voltaje calculados en los preamplificadores 107
TABLA 1.10) Valores calculados de diferentes parámetros del amplificador de potencia 107
TABLA 1.11) Diferentes características medidas y calculadas del amplificador de potencia 107
TABLA 2.1) Relación voltaje - código 128
TABLA 2.2) Características generales de la PCI-DAS 1000 138
TABLA 2.3) Tabla de verdad del 82C55 141
TABLA 2.4) Tabla de verdad del 82C54 143
TABLA 2.5) Tabla de función de cada uno de los terminales del X9C503 148
TABLA 2.6) Valores medidos de la PCI-DAS 1000 150
TABLA 4.1) Equipos utilizados en las mediciones 206
TABLA 4.2) Costo de los elementos que conforman la fuente de poder 208
TABLA 4.3) Costo de los elementos que conforman la etapa de preamplificación 209
TABLA 4.4) Costo de los elementos que conforman el amplificador 209
TABLA 4.5) Costo de los elementos que conforman la etapa de protección 209
TABLA 4.6) Costo de los elementos que conforman el interfaz hardware 210
TABLA 4.7) Costo de los elementos que conforman la etapa de medición 210
TABLA 4.8) Costo de accesorios de presentación del prototipo 210

TABLA 4.9) Costo del sistema de amplifiación de audio controlado por computadora 210

CAPÍTULO 1:
AMPLIFICADORES DE POTENCIA DE AUDIO

1.1 INTRODUCCIÓN

La comunicación no solo se realiza de una persona a otra, existen eventos a los cuales asiste una gran audiencia, razón por la cual, es necesario que la transmisión tenga niveles de audio apropiados para que puedan ser captados por todos los participantes a estos eventos.

Por lo antes mencionado, surge la necesidad de implementar sistemas que amplifiquen la señal generada por el transmisor, para lograr que las personas puedan tener una adecuada percepción de la señal emitida.

El primer paso de este proceso es la conversión de la energía acústica a energía eléctrica, utilizando un transductor (micrófono), para luego realizar el proceso inverso, es decir, convertir la energía eléctrica en energía acústica usando otro transductor (parlante).

Este sistema básico compuesto de un micrófono y un parlante no es el más idóneo, puesto que el micrófono usualmente entrega una señal de voltaje en el orden de los milivoltios, con una impedancia interna de 600 Ohmios por ejemplo, mientras que el parlante, presenta como característica específica, una impedancia de 8 Ohmios y una eficiencia alrededor del 10%, dando como resultado problemas de desacoplamiento de impedancias y una potencia acústica de salida en el orden de los nanovatios.

De lo expuesto anteriormente, el sistema básico descrito, no cumple con la necesidad de amplificar la señal a niveles adecuados de potencia, por lo que se considera la necesidad de incorporar un sistema de amplificación de audio el cual está conformado por cuatro etapas (preamplificador, mezclador, ecualizador y amplificador) como lo demuestra la figura 1.1

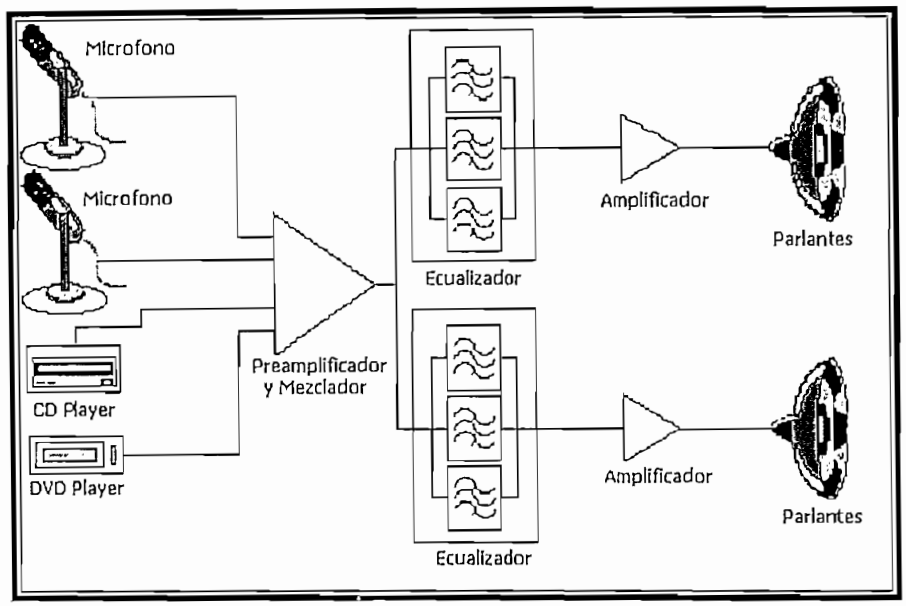


FIGURA 1.1) Partes de un Sistema de Audio

Las entradas son cualquier señal proveniente de un micrófono, casete, CD, DVD, DAT, ADAT, MD, MP3, etc., que están en el rango de los milivoltios. Estas entradas ingresan por una etapa mezcladora (utilizada cuando existe más de una entrada) cuya función está fundamentada en mezclar y amplificar en bajo nivel todas las entradas existentes.

La etapa de ecualización cumple la función de otorgarle una mejor calidad a la señal que se va a obtener a la salida, amplificando o atenuando distintos rangos de frecuencias, de tal manera que cambie la respuesta de frecuencia según el requerimiento.

Se podría prescindir de esta etapa siempre y cuando las necesidades del usuario no las requieran.

La etapa de potencia es la encargada de suministrar a las salidas, en este caso los parlantes, una señal similar a la señal de entrada, pero con mayores niveles de potencia. Figura 1.2.

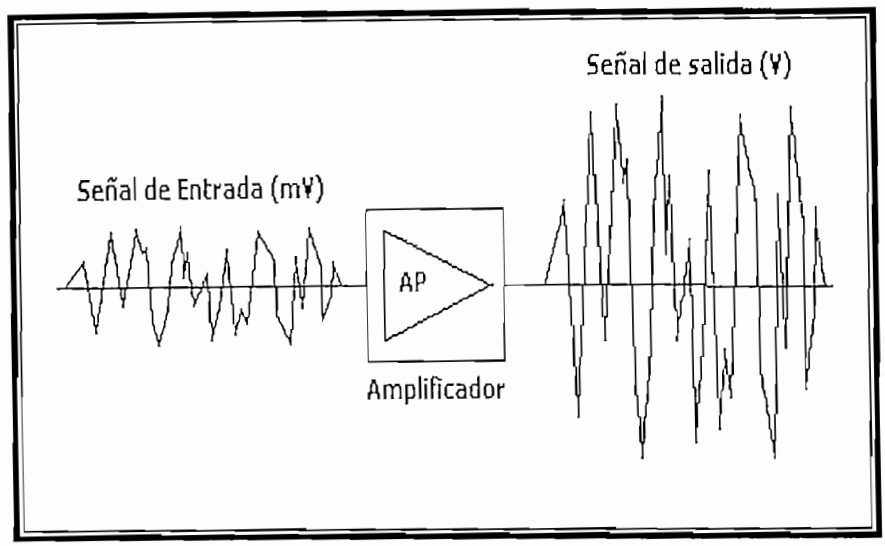


FIGURA 1.2) Proceso de amplificación de una señal

La figura 1.2 representa el proceso que realiza la etapa de potencia, demostrando el aumento de la tensión (V) de la señal sin perturbar la forma de onda, además de suministrar gran cantidad de corriente (I)

1.2 AMPLIFICADORES DE POTENCIA

La función de los amplificadores de potencia consiste en amplificar los valores de corriente y voltaje de cualquier señal que ingresa a éste, para obtener a la salida una señal con mayores niveles de potencia en relación con la que ingresó.

Este aumento de potencia, es el resultado de la energía proporcionada por la fuente de alimentación. Estos circuitos de amplificación para señales AC, son el resultante de combinar valores de AC y DC.

La señal que se obtiene a la salida, tiene igual forma de onda que la señal que se percibió inicialmente a la entrada, pero varían las magnitudes. En lugar de tensiones de decenas de milivoltios (mV), se obtiene a la salida tensiones de decenas de voltios (V) y corrientes de varios amperios (A).

La señal que ingresa al amplificador se mide en milivatios, es decir, tiene una potencia de aproximadamente 1000 veces menos que la que tendrá a la salida.

1.2.1 DIAGRAMA DE BLOQUES

El diagrama de bloques de un amplificador de potencia consta de las siguientes etapas:

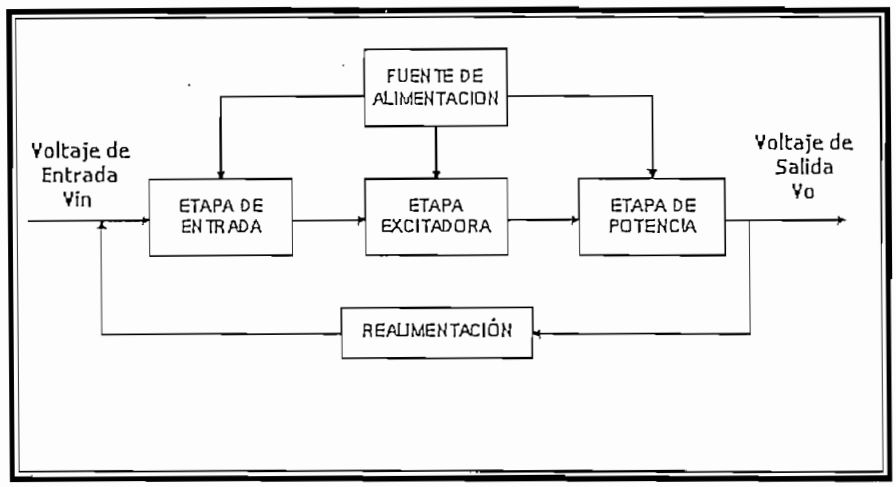


FIGURA 1.3) Diagrama de bloques de una amplificador de potencia

1.2.1.1 Etapa De Entrada

Es el punto al cual llega la señal de entrada. Esta sección define la impedancia de entrada del conjunto (Amplificador de Potencia), además la tensión de la señal de entrada experimenta un mínimo aumento antes de dirigirse a la etapa excitadora.

Puede estar constituida por alguna clase de amplificador de voltaje, tal como un amplificador diferencial, cuya principal característica es tener una alta impedancia de entrada y muy baja distorsión, aunque su ganancia de voltaje no sea tan elevada.

Al poseer una alta impedancia de entrada, proporciona la independencia entre el circuito de realimentación y el módulo precedente al amplificador de potencia, evitando cualquier tipo de influencia sobre la unidad anterior, como puede ser el ecualizador.

1.2.1.2 Etapa Excitadora

Es la encargada de "excitar" la etapa de potencia. Es decir recibe una señal de pocos milivoltios y la amplifica, de tal manera que se eleva el nivel de voltaje antes de pasar a la etapa de potencia.

La amplificación de potencia es dada por la adecuada amplificación de la corriente y del voltaje de la señal de trabajo. Por lo general las etapas de potencia se encargan de amplificar la señal de corriente y mantener la ganancia de voltaje.

Por tal razón la Etapa Excitadora provee la amplificación de voltaje adecuada para entregarla a la etapa de potencia y obtener el incremento de potencia requerida.

1.2.1.3 Etapa De Potencia o De Salida

Esta etapa tiene una alta importancia, razón por la cual, otorga el nombre a todo el conjunto. Ésta es la encargada de dotar de potencia a la señal de salida, puesto que la señal que recibe tiene mucho voltaje, pero muy poca intensidad de corriente.

Esta etapa es la que proporciona varios amperios de intensidad de corriente eléctrica a la señal, sin embargo, no aumenta el voltaje de la etapa anterior.

En esta etapa se manejan tensiones y corrientes muy elevadas, por tal motivo, es la que más recursos energéticos demanda de la fuente de alimentación; es decir, que ésta es la etapa donde existe mayor consumo de energía.

1.2.1.4 Realimentación

La realimentación es un proceso por el cual un circuito realimentador toma una porción de la señal de salida y la lleva a la entrada, esta técnica hace a los circuitos electrónicos más útiles, flexibles y confiables.

La realimentación, también es conocida como retroalimentación y puede ser de dos tipos: Positiva y Negativa.

La positiva es utilizada en circuitos osciladores y la negativa es utilizada para mejorar la estabilidad de los amplificadores electrónicos.

Algunas de las ventajas que ofrece la retroalimentación negativa son las siguientes:

- Estabilizar la ganancia y mejorar la confiabilidad del rendimiento del circuito.
- Lograr un control preciso de la ganancia y niveles de impedancia terminal adecuados.
- Sobreponer las propiedades no ideales de los componentes físicos, cuyos parámetros pueden variar con el tiempo de trabajo, el voltaje de la línea, la temperatura o el reemplazo.
- Reducir la distorsión no lineal, y por tanto, aumentar la variación de la señal de entrada permitida.
- Cambiar las características dinámicas, tales como la respuesta de amplitud, frecuencia y fase, etc. del amplificador.

Por la gran utilidad de la retroalimentación negativa, se realiza un breve detalle acerca de su funcionamiento.

El diagrama de bloques representativo de un circuito Retroalimentado negativamente es el siguiente:

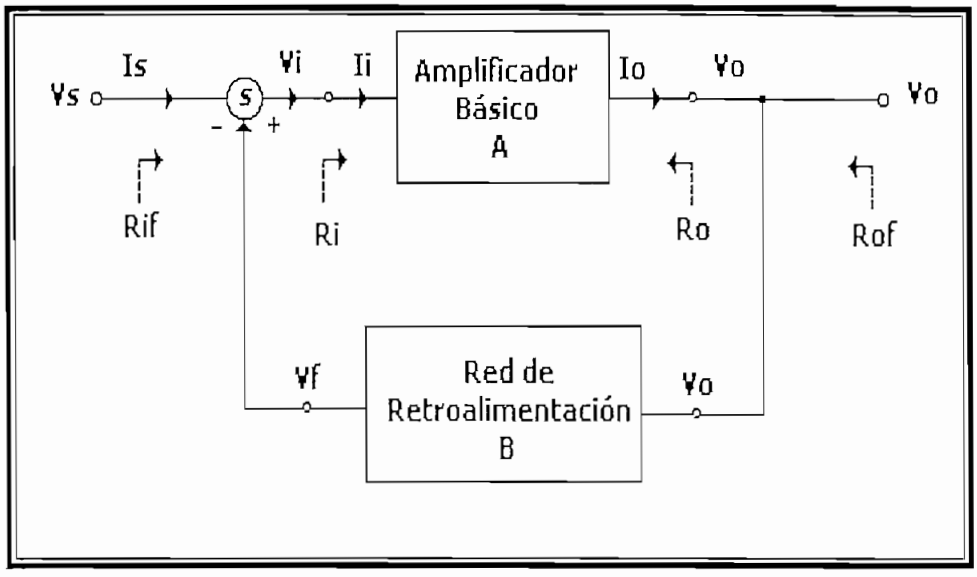


FIGURA 1.4) Diagrama de bloques de un circuito con Retroalimentación Negativa

El sistema Retroalimentado que se puede observar en el diagrama anterior, es la representación gráfica de un circuito amplificador básico con ganancia A, que tiene conectado entre sus terminales de salida y de entrada un circuito de retroalimentación con ganancia B.

Se debe considerar que las señales se transmiten únicamente en la dirección de las flechas.

El circuito amplificador básico, cuya ganancia de voltaje es la relación del voltaje de salida sobre el voltaje de entrada, es igual a:

$$A_V = \frac{V_o}{V_i} \qquad \text{Ec. 1.1}$$

Mientras que la red de retroalimentación tiene una ganancia de:

$$B_V = \frac{V_f}{V_o} \qquad \text{Ec. 1.2}$$

En la entrada del amplificador básico hay un circuito sumador «*S*», el mismo que otorga una señal de salida V_i o I_i proporcional a la suma de las señales de la fuente y del circuito de retroalimentación, como lo demuestra la figura 1.4.

Si se considera que la señal de retroalimentación V_f, tiene una polaridad contraria a la de V_S, se puede determinar que la señal resultante en la entrada del amplificador básico es:

$$V_s = V_i + V_f \qquad \text{Ec. 1.3}$$

De las ecuaciones anteriores, se puede claramente determinar que la ganancia total con retroalimentación A_T seria igual a:

$$A_{Vf} = \frac{V_o}{V_s} = \frac{V_o}{V_i + V_f} \qquad \text{Ec. 1.4}$$

Dividiendo tanto el denominador como el numerador para Vi:

$$A_{Vf} = \frac{\frac{V_o}{V_i}}{\frac{V_i}{V_i} + \frac{V_f}{V_i}} \qquad \text{Ec. 1.5}$$

Se puede definir que la ganancia de voltaje del circuito Retroalimentado (A_T), en función de las ganancias antes mencionadas, es igual a:

$$A_{Vf} = \frac{A_V}{1 + B_V A_V} \qquad \text{Ec. 1.6}$$

La ecuación anterior es la ecuación básica de la retroalimentación negativa y puede llegarse a ella independientemente de que las cantidades de entrada y salida sean voltajes o corrientes. Vale mencionar que el valor de $|A_{Vf}|$ es menor que el valor de $|A_V|$ en este tipo de retroalimentación.

Para el caso de retroalimentación positiva; es decir, con $V_s = V_i - V_f$, se tiene que:

$$A_{Vf} = \frac{A_V}{1 - B_V A_V} \qquad \text{Ec. 1.7}$$

Así para la retroalimentación positiva el valor de $|A_{Vf}|$ es mayor que $|A_v|$.

Se tienen cuatro diferentes tipos de retroalimentación negativa, y cada uno presenta efectos diferentes, tanto en la impedancia de salida como en la impedancia de entrada.

Estos tipos de retroalimentación se muestran en la figura 1.5.

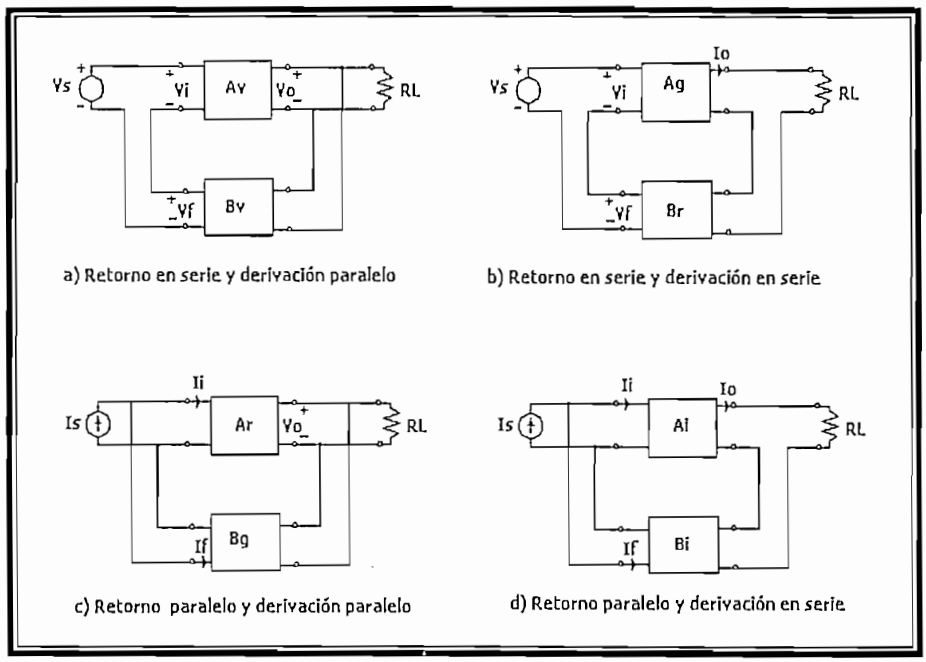


FIGURA 1.5) Diagrama de bloques de los tipos de realimentación negativa

Se denomina como retorno a la parte de la señal de salida que ha sido realimentada a la entrada pudiendo estar en serie o en paralelo con respecto a la señal de entrada. De la misma manera se denomina como derivación a la parte de la señal de salida que se toma para que sea realimentada a la entrada, la derivación pude darse en serie o en paralelo con respecto a la carga RL.

- La figura 1.5.a, permite visualizar un amplificador Retroalimentado con retorno en serie y derivación en paralelo (RSDP), la señal de retroalimentación V_f es proporcional a la señal de salida V_o.

  La conexión paralelo en la salida reduce a R_o. La conexión serie en los terminales de entrada aumenta a R_i. El amplificador resultante es un verdadero amplificador de voltaje y el factor de retroalimentación en voltaje es:

$$B_V = \frac{V_f}{V_o} \qquad \text{Ec. 1.8}$$

- En un amplificador retroalimentado con retorno en serie y derivación en serie (RSDS), como se muestra en la figura 1.5.b, las conexiones en serie de los terminales de salida y de entrada aumentan tanto a R_o como a R_i.

  El amplificador resultante es un amplificador del tipo de transconductancia y el factor de retroalimentación transitivo es:

$$B_r = \frac{V_f}{I_o} \qquad \text{Ec. 1.9}$$

- En un amplificador con retorno paralelo y derivación en paralelo (RPDP), como se muestra en la figura 1.5.c, la señal retroalimentada I_f es proporcional a la señal de salida V_o.

  Las conexiones paralelo en los terminales de salida y de entrada reducen tanto a R_o como a R_i.

  El amplificador resultante es una amplificador del tipo transresistencia y el factor de retroalimentación transconductivo es:

$$B_g = \frac{I_f}{V_o} \qquad \text{Ec. 1.10}$$

- En un amplificador Retroalimentado con retorno en paralelo y derivación en serie (RPDS), figura 1.5.d, la conexión serie en la salida aumenta R_o y la conexión paralelo en la entrada reduce a R_i.

El amplificador resultante es un verdadero amplificador de corriente y el factor de retroalimentación de corriente es:

$$B_i = \frac{I_f}{I_o} \qquad \text{Ec. 1.11}$$

A continuación se va a detallar los valores de impedancia de entrada y los de impedancia de salida para un circuito con retroalimentación RSDP. Posteriormente se lo hará con un amplificador de voltaje *No inversor*, que es un claro ejemplo de un circuito con retroalimentación RSDP que trabaja a partir de amplificadores operacionales y que será utilizado en el diseño de los preamplificadores.

- **Impedancia de entrada (Z_{in})**

A la entrada de un circuito con retroalimentación RSDP se obtiene el siguiente circuito:

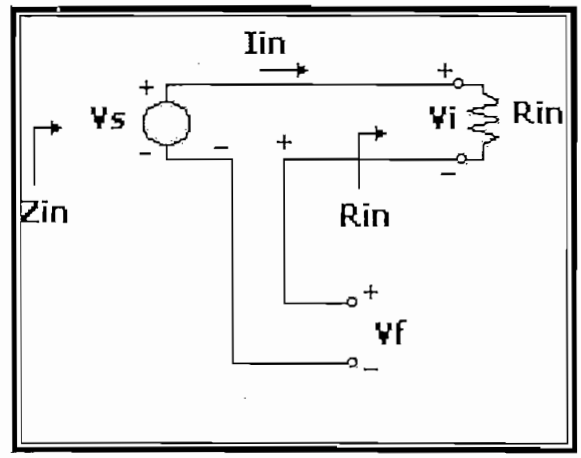


FIGURA 1.6) Diagrama de bloques de la entrada de un circuito con realimentación RSDP

En donde: Z_{in} = Impedancia de entrada del circuito con retroalimentación

R_{in} = Impedancia de entrada del amplificador básico sin Retroalimentación.

Se tiene las siguientes expresiones conocidas:

$V_s = V_i + V_f$ (asumiendo que R_S tiende a cero)

$V_i = R_{in}.I_{in}$

$V_o = V_i.A_v$

$V_f = B_v.V_o \rightarrow V_f = A_v.B_v.V_i$

La impedancia de entrada del circuito con retroalimentación esta dada por:

$$Z_{in} = \frac{V_s}{I_{in}} \qquad \text{Ec. 1.12}$$

Con lo que reemplazando las expresiones antes citadas se tiene que:

$$Z_{in} = \frac{V_i + V_f}{I_{in}} \qquad \text{Ec. 1 13}$$

$$Z_{in} = \frac{V_i + V_i \cdot B_v \cdot A_v}{I_{in}} \qquad \text{Ec. 1.14}$$

$$Z_{in} = \frac{V_{in}\left(1 + B_v \cdot A_v\right)}{I_{in}} \qquad \text{Ec. 1.15}$$

$$Z_{in} = R_{in}\left(1 + B_v \cdot A_v\right) \qquad \text{Ec. 1.16}$$

Como se tiene que el factor $(1+B_vA_v)$ es mayor que 1, el valor de Z_{in} siempre será mayor que R_{in}.

$$Z_{in} \rangle R_{in} \qquad \text{Ec. 1.17}$$

- Impedancia de Salida (Z_o)

Para encontrar la impedancia de salida, se cortocircuita la entrada es decir que $V_s = 0$ y se excitan los terminales de salida con una fuente de voltaje V_x como se demuestra en la figura 1.7. Además se continúa asumiendo que la resistencia de la fuente R_s tiende a cero.

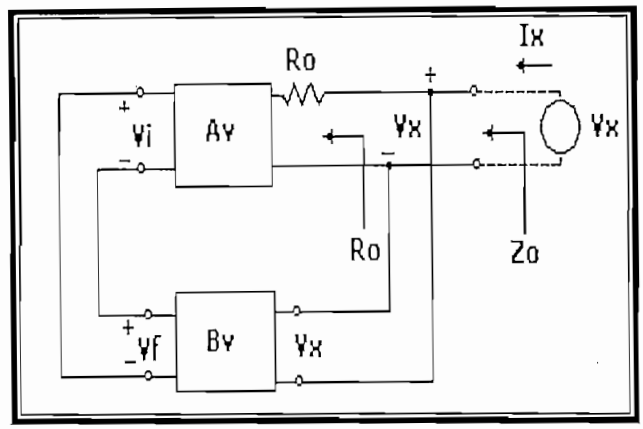


FIGURA 1.7) Diagrama de bloques de la salida de un circuito con retroalimentación RSDP

Según el circuito anterior se tienen que:

$$V_i = -V_f = -B_v . V_x \qquad \text{Ec.1.18}$$

Para encontrar la impedancia de salida del circuito se aplica la siguiente expresión:

$$Z_o = \frac{V_x}{I_x} \qquad \text{Ec.1.19}$$

En donde asumiendo que la impedancia R_o es mucho menor que la resistencia que se ve del circuito retroalimentador, I_x es igual:

$$I_x = \frac{V_x - (-V_f) . A_v}{R_o} \qquad \text{Ec. 1.20}$$

$$I_x = \frac{V_x + B_v \cdot A_v \cdot V_x}{R_o} \qquad \text{Ec. 1.21}$$

La impedancia de salida que se genera a la salida de un circuito con retroalimentación RSDP sería:

$$Z_o = \frac{R_o}{1 + A_v B_v} \qquad \text{Ec. 1.22}$$

Como el factor $1 + A_v B_v$ es mayor que 1, se obtiene como resultado que la impedancia de salida de un circuito Retroalimentado es menor que la de un circuito sin retroalimentación.

1.2.1.5 Fuente De Alimentación

Es un dispositivo que adapta la energía de la red eléctrica general (110 - 120 V_{AC}), para que pueda ser usada por las distintas etapas anteriormente mencionadas.

Las fuentes de alimentación suelen ser simétricas, es decir, fuentes que entregan un mismo voltaje de DC pero de diferente polaridad. La energía entregada por la fuente de alimentación, debe ser suficientemente grande para poder abastecer a la etapa de potencia de toda la energía que ésta requiera.

Una etapa de potencia estéreo tiene que duplicar las tres etapas (entrada, excitadora y potencia) y puede utilizar una fuente de alimentación para todos.

Un diagrama de una fuente simétrica básica se muestra en la figura 1.8.

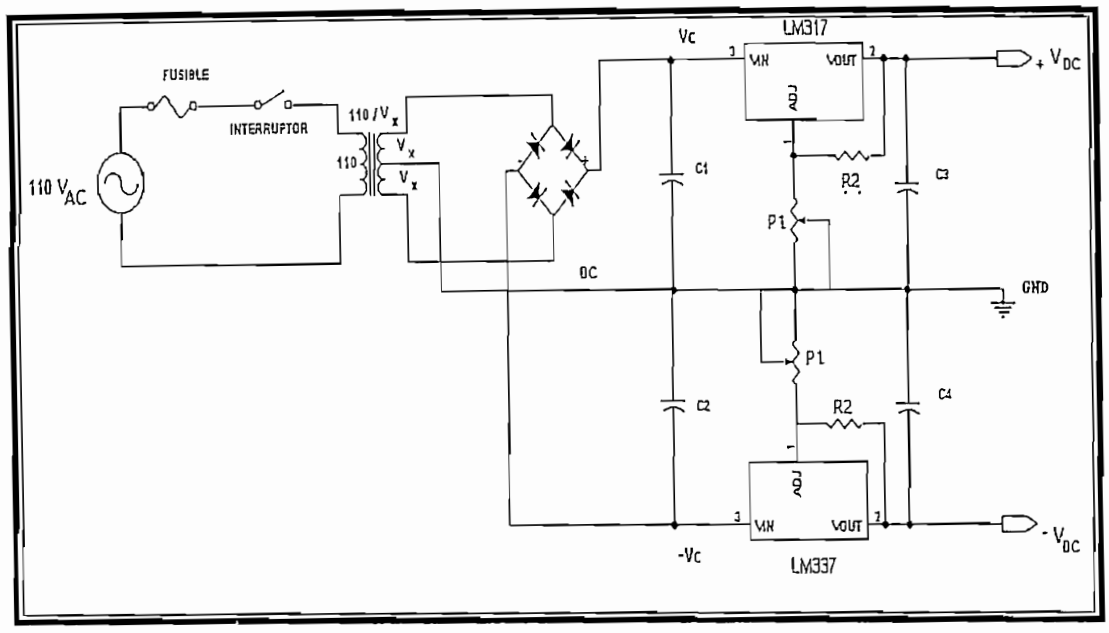


FIGURA 1.8) Diagrama de una Fuente de Alimentación DC variable

Como se demuestra en la figura es necesario contar con un transformador de toma central (transformador con su secundario dividido en dos salidas de voltaje AC iguales). El valor del voltaje de salida (rms) que se da a las subdivisiones del secundario dependen del valor de voltaje V_c que se quiera obtener y de los criterios de diseño, se lo calcula con la siguiente expresión aproximada:

$$Vx_{rms} \approx \frac{\pi . V_c}{2 . \sqrt{2}} \qquad \text{Ec. 1.23}$$

Las variaciones que se tenga en el valor del voltaje del secundario del transformador, debido a las fluctuaciones en la red de alimentación eléctrica (100V – 120V) deben ser tomadas en cuenta, por lo cual se debe tomar un valor de voltaje V_c que salvaguarde las condiciones que se requieren. Sin embargo la utilización de CI´s reguladores de voltaje nos ayudan a pasar por alto estas fluctuaciones (de acuerdo al rango permitido), ya que estos permiten un cierto rango de voltaje de entrada para entregar un voltaje "fijo" a la salida.

El puente de diodos se lo escoge de acuerdo a la corriente máxima con la que se va a trabajar y que debería entregar la fuente.

Los capacitores C_1 y C_2 son los encargados de convertir a la señal rectificada por el puente de diodos en una señal que tienda a ser constante. Mientras más grande es el valor de estos capacitores el tiempo de descarga es mayor y se obtienen mejores resultados en la conversión.

Los circuitos Integrados LM317 y LM337 son reguladores de voltaje variables tanto positivo como negativo respectivamente, los cuales permiten controlar el voltaje DC que se entrega a la salida entre un rango de ±1.2 a ±37 Voltios y que soportan una corriente máxima de 1.5 Amperios. También existe otro tipo de regulares de voltaje pero estos son fijos pertenecen a la serie 78XX para voltajes positivos y 79XX para voltajes negativos, XX representa el voltaje DC que entregan y cuyos valores están estandarizados[1].

Por ejemplo el 7818 es un regulador de voltaje positivo de +18 Voltios que permite un rango de voltajes a la entrada de +21V a +33V, de igual manera el 7918 es un regulador de voltaje negativo de –18 Voltios que permite un rango de voltajes a la entrada de –21V a –33V. Estos integrados así mismo soportan una corriente máxima de 1 A.

Los capacitores C_3 y C_4 se los utiliza para eliminar algún tipo de ruido que se presente y sus valores son bajos y para tener R_o de la fuente bajo para altas frecuencias.

Si se desea implementar una fuente solamente positiva o negativa se realizaría una sola parte de circuito. El diagrama y diseño de la fuente implementada en este proyecto se lo presenta posteriormente.

---

[1] La información de estos valores se encuentra en el Anexo C.

1.3 CLASES DE AMPLIFICADORES DE POTENCIA

La clase de operación de los amplificadores está definida por el porcentaje de la señal de entrada que pueden reproducir a la salida. Cada una de estas clases tiene su rendimiento como característica determinante.

Actualmente se conocen otras clases de amplificadores, diferentes a aquellas que comúnmente se solían tener (A, B, AB y C), como son las clases D, E, G y H, las mismas que ya no son clasificadas de la misma manera como se hizo con las primeras. La tabla 1.1 demuestra la diferencia entre las clases de amplificadores (A, B, AB y C), según el ciclo de operación de la señal de salida y el rendimiento.

CLASE	CICLO DE OPERACIÓN	RENDIMIENTO
A	360°	25% a 50 %
B	180°	78.5%
AB	180° A 360°	Entre 25% (50%) y 78.5%
C	Menor a 180° (típicamente 120°)[2]	Elevado rendimiento, pero elevada distorsión

TABLA 1.1) Rendimiento y ciclo de operación de las diferentes clases de amplificadores

1.3.1 CLASE A

Esta clase usualmente se la implementa en etapas de excitación por su baja distorsión en la señal, no se implementa en etapas de potencia debido a que proporciona poca potencia y bajo rendimiento. En esta clase el componente activo utilizado (TBJ, FET, MOSFET, etc.), trabaja sobre la región lineal, con la particularidad de que el punto Q (punto de trabajo para el caso del transistor) se ubica aproximadamente en la mitad de la línea de carga, como se muestra en la figura 1.9, de tal manera que la señal de entrada pueda variar hacia arriba y abajo

[2] Tomado del libro: Grob Basic Electronics, 1992, 7ma edición, Capitulo 30, página 762

sin sobrepasar la región de corte y/o saturación, obteniéndose como resultado un ángulo de conducción de 360° en la señal de salida.

La desventaja que genera esta clase de amplificador, es el consumo permanente de corrientes altas de la fuente de alimentación, independientemente de si existe o no señal, por tal razón presenta el inconveniente de generar una fuerte y constante emisión de calor.

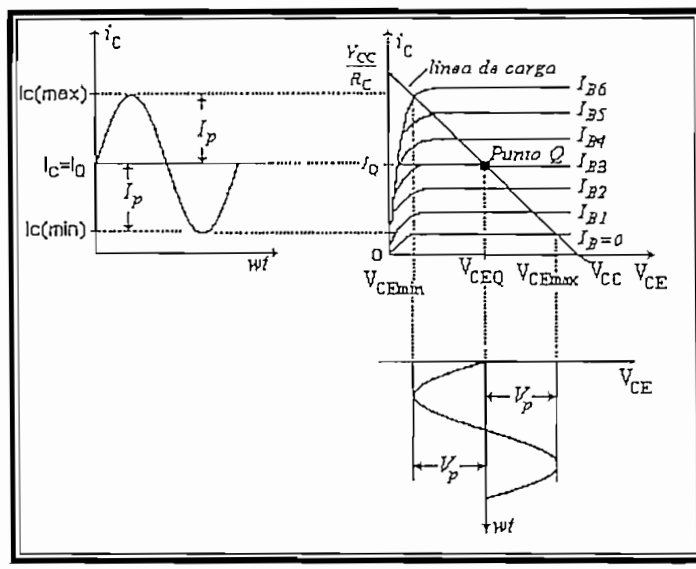


FIGURA 1.9) Ciclo de operación de un amplificador Clase A

1.3.2 CLASE B

En esta segunda clase de amplificación se tiene la particularidad de no disponer de corriente a través de los transistores, si no existe una señal presente. La propia señal de entrada polarizará a los transistores para que entren en conducción y exciten a su vez a la carga.

El funcionamiento de este tipo de amplificadores, es con dos componentes activos, los mismos que trabajan en región de corte de la siguiente manera: el un componente se usa para conducir en los semiciclos positivos de la señal y el otro para los semiciclos negativos (a esta configuración se la conoce *como push pull* o de simetría complementaria) Figura 1.10

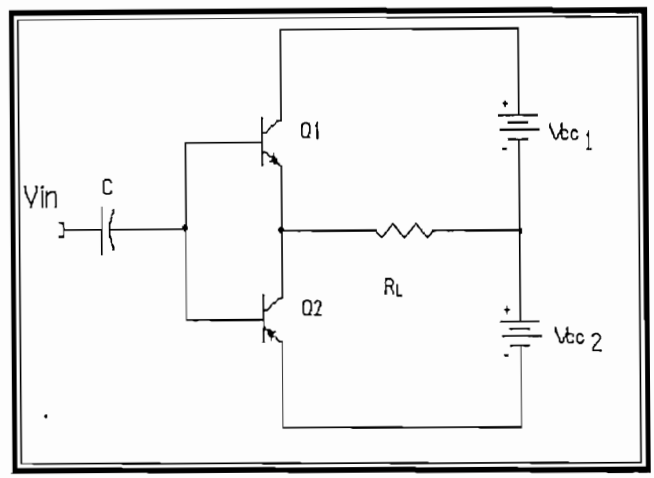


FIGURA 1.10) Circuito en contrafase de simetría complementaria

Para el caso de una señal sinusoidal se tiene un componente respondiendo a cada semiciclo, la operación demandada es la mitad del tiempo, de modo que la corriente de cada uno es diferente de cero en el 50% del tiempo, razón por la cual esta configuración no permite que se consuma energía de la fuente de alimentación sino se tiene señal de entrada (reposo).

Por otro lado la desventaja de esta configuración es la distorsión que se produce a la salida en el cruce por cero (figura 1.11), esta reacción se produce debido a que los componentes activos trabajan en la región de corte como se indicó anteriormente, en la cual no se puede ni predecir ni controlar el comportamiento del componente.

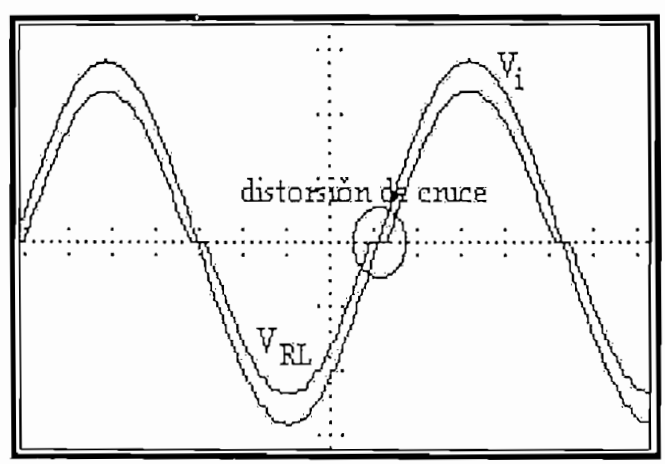


FIGURA 1.11) Distorsión en el cruce por cero en un amplificador clase B

1.3.3 CLASE AB

Es similar a un amplificador clase B pero con un pequeño nivel de voltaje en la polarización de los componentes activos para disminuir la distorsión de cruce, aunque su resultante es el aumento en el consumo de energía.

Para DC, un componente activo en clase AB tiene una corriente de base por encima del nivel de corriente de base de la clase B ($I_b = 0$) y no sobrepasa la mitad de la corriente de base de la clase A, esta es la razón para que este tipo de amplificador tome este nombre.

El ángulo de conducción de corriente en estos amplificadores está entre los 180° y 360°, por lo que se requiere una conexión push pull o contrafase (figura 1.10) para lograr un ciclo de salida completo.

1.3.4 CLASE C

Los amplificadores de clase C son similares a los de clase B debido a que el componente activo tiene una corriente de base igual a cero.

Las desventajas de los amplificadores de clase B son más evidentes en los amplificadores de clase C (por ejemplo la distorsión), por tanto los de clase C no son prácticos para audio. Esta clase de amplificadores tiene un gran rendimiento pero también una elevada distorsión.

En un amplificador clase C la corriente de salida fluye por debajo del medio ciclo, típicamente trabaja a 120 °.

Los amplificadores en clase C no están orientados para ser amplificadores de audio ya que trabajan una frecuencia específica (depende del circuito

resonante). Un amplificador clase C, como se muestra en la figura 1.12, está polarizado para operar en menos de 180° del ciclo de señal de entrada. Sin embargo, el circuito sintonizado de la salida proporcionará un ciclo completo de la señal de salida para la frecuencia fundamental o resonante del circuito sintonizado de la salida.

Este tipo de operación está limitado para usarse a una frecuencia fija, por ejemplo, en un circuito de comunicaciones.

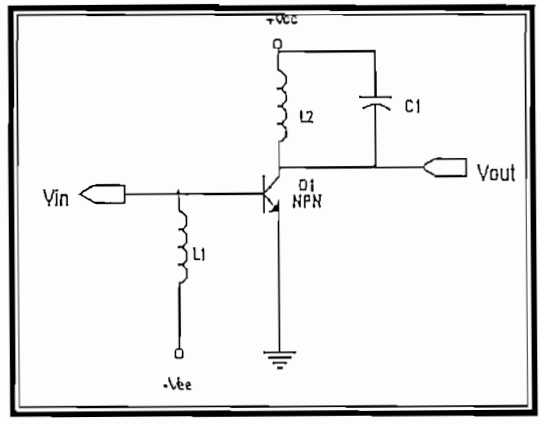


FIGURA 1.12) Circuito de potencia Clase C

1.3.5 CLASE D

El amplificador de clase D, está diseñado para trabajar con señales digitales o de tipo pulso, logrando una eficiencia superior al 90%. Su funcionamiento es diferente a todas las clases detalladas anteriormente.

En esta clase, es necesario convertir cualquier señal de entrada a una forma de onda tipo pulso antes de usarla para manejar una carga de potencia grande, y posteriormente convertir a la señal de retorno a tipo senoidal para recuperar la señal original.

Para esta conversión se debe modular a la señal de entrada con una onda cuadrada (o diente de sierra) como portadora, luego se amplifica la señal

modulada y finalmente se filtra en paso bajo la señal amplificada para volver a la señal de banda base como se muestra en la figura 1.13.

La clase D, se distingue por el mejoramiento en su rendimiento y la nueva filosofía de trabajo. También se la conoce como amplificación digital.

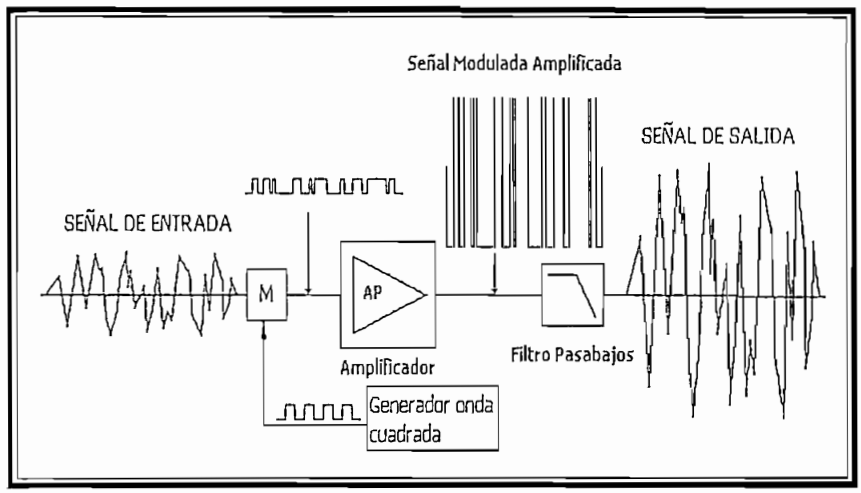


FIGURA 1.13) Diagrama explicativo del funcionamiento de un Amplificador Clase D

Sin embargo, aunque los amplificadores clase D son buenos por su eficiencia, no obstante son desastrosos por otras razones.

Es esencial que un amplificador clase D esté acompañado por un filtro paso-bajo para eliminar el ruido de conmutación. Este filtro añade distorsión y desplazamiento de fase, inclusive limita las características del amplificador en alta frecuencia, y es raro que los amplificadores de clase D tengan buenos agudos. El mejor uso para estos amplificadores es en los sub-bajos.

1.3.6 CLASE E

Este amplificador es un amplificador de pulsos (su rendimiento puede ser muy elevado) cuya salida se encuentra sintonizada a una determinada frecuencia. Suele ser empleado en aplicaciones de radio cuando se trabaja a una única

frecuencia o bien en un margen muy estrecho de frecuencias. No es útil para aplicaciones de audio.

1.3.7 CLASE G

La clase G se refiere a amplificadores conmutados que tienen dos diferentes fuentes de alimentación. La fuente para el amplificador se conecta al voltaje menor para señales débiles y al voltaje mayor para señales fuertes, esto permite obtener más eficiencia sin requerir la conmutación de las etapas de salida, de tal manera que pueden amplificar las señales de audio con mejores niveles que los amplificadores de clase D.

1.3.8 CLASE H

La clase H se basa en utilizar un amplificador en clase D o una fuente de alimentación conmutada para alimentar a un amplificador en clase AB o A. De este modo el amplificador presenta un excelente rendimiento y emana el sonido de un buen amplificador clase AB. La clase H es la más utilizada en etapas de equipos profesionales.

1.4 PARÁMETROS CARACTERÍSTICOS DE LOS AMPLIFICADORES DE POTENCIA

Entre los principales parámetros característicos de los amplificadores de potencia están los siguientes:

1.4.1 POTENCIA ENTREGADA POR LA FUENTE DE ENERGÍA

Es la potencia entregada por las fuentes de energía al amplificador.

La potencia será:

$$P_{DC} \cong V_{CC} . I_{DC} \qquad \text{Ec. 1.24}$$

Donde I_{DC} es la corriente que entrega la fuente de energía (corriente continua). Para el caso de un amplificador clase B el consumo de corriente de una sola fuente de alimentación tiene la forma de una señal rectificada de media onda. El valor I_{DC} puede encontrarse con la siguiente expresión:

$$I_{DC} = \frac{2}{\pi} . I_p \qquad \text{Ec. 1.25}$$

I_P es el valor pico de la forma de onda de la corriente, con lo que la ecuación 1.24 se representaría de la siguiente manera:

$$P_{DC} = V_{CC} . \frac{2}{\pi} . I_P \qquad \text{Ec. 1.26}$$

1.4.2 POTENCIA ÚTIL ENTREGADA A LA CARGA

Conocida como P_{AC}, es la potencia entregada a la carga por el sistema amplificador. Si la señal de salida es de forma senoidal y tomando su valor rms, la potencia útil entregada a la carga será:

$$P_{AC} = \frac{V_{RL}^2 (\text{rms})}{R_L} \qquad \text{Ec. 1.27}$$

1.4.3 RENDIMIENTO DEL AMPLIFICADOR

Se define como rendimiento del amplificador, a la relación entre la potencia entregada a la carga(P_{AC}) y la potencia entregada por la fuente de alimentación (P_{DC}) del sistema de amplificación.

$$\eta = \frac{P_{AC}}{P_{CC}}.100\ \% \qquad \text{Ec. 1.28}$$

Para la clase B, la potencia máxima de salida es entregada a la carga cuando se tiene que: $V_{RL} = V_{CC}$ y $I_P = V_{CC} / R_L$, con lo que se tiene una eficiencia máxima al 78.54 %.

1.4.4 DISTORSIÓN

Cualquier amplificador ideal es capaz de amplificar una señal pura para proporcionar una versión mucho más grande de está. Cuando sucede una distorsión, la salida no será un duplicado exacto (a excepción de magnitud) de la señal de entrada.

La distorsión es producida por las características no lineales de los dispositivos con los cuales se trabaja, en cuyo caso sucede la distorsión no lineal o de amplitud.

La distorsión también puede suceder debido a que los elementos del circuito y los dispositivos responden en forma diferente a la señal de entrada de las diferentes frecuencias, siendo ésta una distorsión por frecuencia.

1.4.4.1 Distorsión armónica

Este tipo de distorsión es a la que más cuidado se la debe prestar, en lo que es sistemas de audio.

Se considera que una señal tiene distorsión cuando hay componentes de frecuencia armónicos, es decir componentes de frecuencias múltiplos enteros de la frecuencia fundamental o de trabajo.

Si la frecuencia fundamental tiene una amplitud A_1, y el enésimo componente de frecuencia tiene una amplitud A_n, la distorsión armónica puede definirse como:

$$\%\ \text{de distorsión armónica enésima} = \%\ D_n = \frac{|A_n|}{|A_1|} x100\ \% \qquad \text{Ec. 1.29}$$

En donde, por lo general, el componente fundamental es mucho mayor que cualquier componente armónico.

1.4.4.1.1 Distorsión Armónica Total (THD)

Cuando una señal de salida tiene varios componentes de distorsión armónica individuales, se puede considerar que la señal tiene una distorsión armónica total basada en los elementos individuales combinados por la relación siguiente:

$$\%\ THD = \sqrt{D_2^2 + D_3^2 + D_4^2 + ...}\ x\ 100\ \% \qquad \text{Ec. 1.30}$$

1.5 AMPLIFICADORES DE AUDIO

Los amplificadores de potencia de audio deben entregar desde 100 mW hasta pocos cientos de vatios de potencia para activar algún dispositivo, como por ejemplo un parlante, las consideraciones fundamentales son: potencia AC de salida, ganancia de potencia de la etapa y la potencia de excitación necesaria de la etapa precedente, eficiencia de la conversión de potencia DC a AC, disipación

de potencia en el dispositivo activo, distorsión de salida y requisitos de alimentación DC.

Los amplificadores de audio son amplificadores de potencia, con la particularidad de buscar la mínima distorsión en la reproducción de una señal de entrada, con los mayores niveles de potencia posible.

Según estas características lo que usualmente se ha adoptado son los amplificadores en contrafase cuasi-complementario, con algunas modificaciones que se han dictado según la práctica.

1.5.1 AMPLIFICADOR DE AUDIO EN CONTRAFASE CUASI-COMPLEMENTARIO

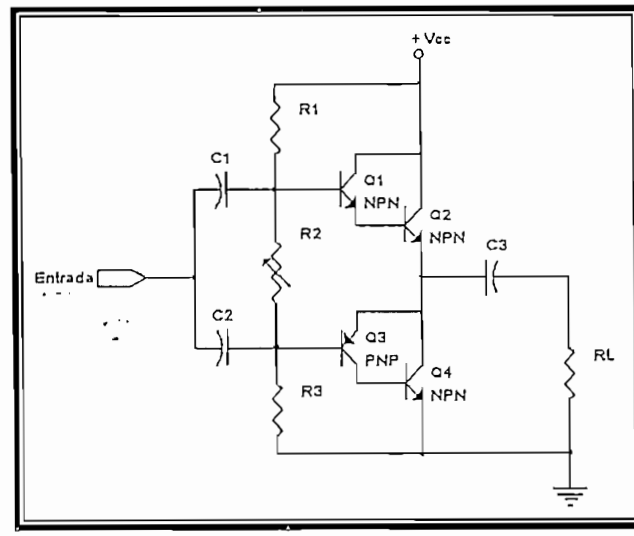


FIGURA 1.14) Amplificador de audio cuasi-complementario

El circuito en contrafase cuasi-complementario, representado en la figura 1.14, es la forma más común de un amplificador de potencia para audio.

Para obtener el circuito contrafase cuasi-complementario, se utiliza dos pares de transistores, el uno como Par Darlington operando en el semiciclo positivo y el otro como Par Retroalimentado que trabaja en el semiciclo negativo

de la señal, realizando así la operación complementaria. La particularidad de estos dos pares es su baja impedancia de salida.

La resistencia R_2 puede regularse para minimizar la distorsión en el cruce por cero mediante el ajuste de la condición de polarización, las resistencias R_1 y R_3 sirven para polarizar a los pares.

1.5.1.1 Par Darlington

Un par Darlington consta de dos transistores bipolares similares (PNP-PNP ó NPN-NPN) conectados en cascada. De la conexión se forma un "transistor compuesto" que trabaja como una sola unidad, con una ganancia de corriente, que es el producto de las ganancias de corriente de los transistores individuales que a su vez está dado por los betas de estos: $\beta_D = \beta_1\beta_2$. Figura 1.15

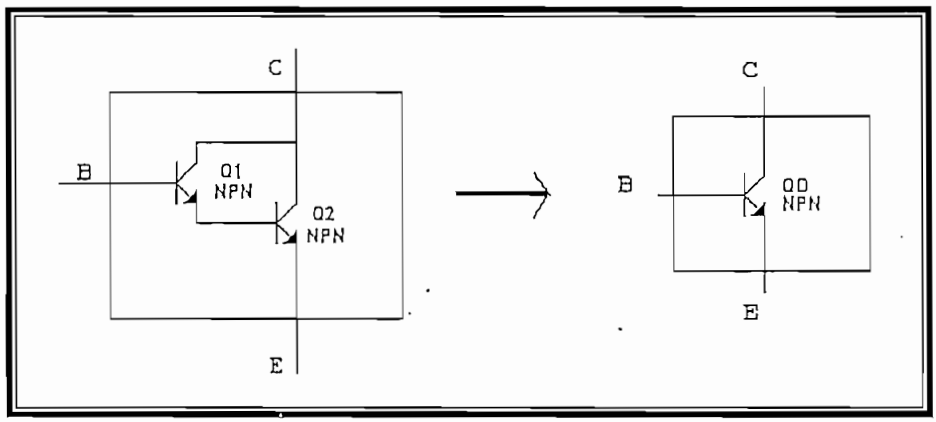


FIGURA 1.15) Circuito par Darlington

- **Impedancia de Entrada (Z_i)**

Para encontrar la expresión que represente la impedancia de entrada de un circuito Darlington, se usa como ayuda el circuito de polarización representado en la figura 1.16 y el circuito AC equivalente, graficado en la figura 1.17.

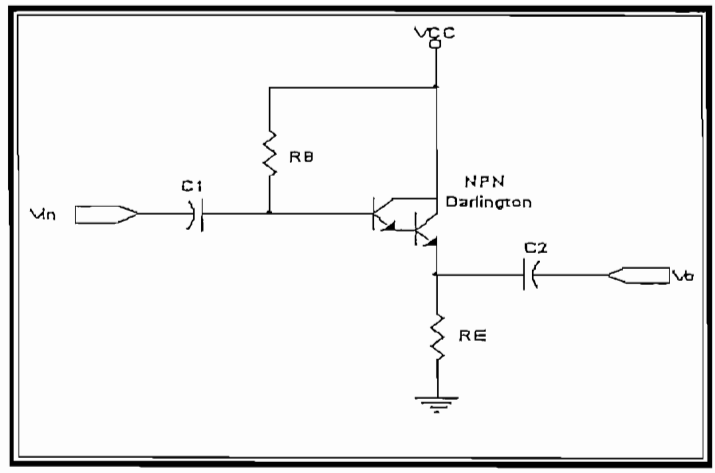


FIGURA 1.16) Circuito amplificador con par Darlington

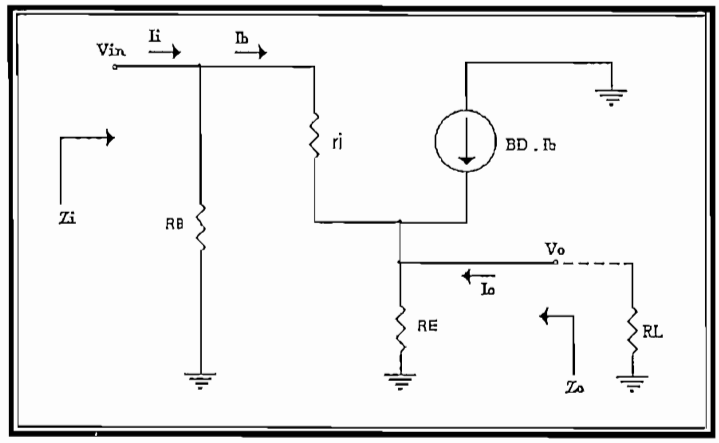


FIGURA 1.17) Circuito AC equivalente de la figura 1.16

Según la figura 1.17 se tiene que la corriente de base (I_b) a través de r_i es:

$$I_b = \frac{V_i - V_o}{r_i} \qquad \text{Ec. 1.31}$$

El voltaje de salida es:

$$V_o = (I_b + \beta_D . I_b) . R_E \qquad \text{Ec. 1.32}$$

Reemplazando se obtiene:

$$I_b . r_i = V_i - V_o = V_i - I_b . (1 + \beta_D) . R_E \qquad \text{Ec. 1.33}$$

Despejando V_i se obtiene:

$$V_i = I_b.[r_i + (1+\beta_D).R_E] \approx I_b(r_i + \beta_D.R_E) \qquad \text{Ec. 1.34}$$

La impedancia que se ve a la entrada del transistor Darlington será:

$$\frac{V_i}{I_b} = r_i + \beta_D.R_E \qquad \text{Ec. 1.35}$$

Viendo la impedancia a la entrada del circuito Z_i:

$$Z_i = R_B \parallel (r_i + \beta_D.R_E) \qquad \text{Ec. 1.36}$$

- **Impedancia de Salida (Z_o)**

Para determinar la impedancia de salida, también se debe basar en el circuito equivalente de la figura 1.17. Para encontrar la impedancia de salida vista por la carga (R_L) se aplica un voltaje V_x a la salida y se cortocircuita la fuente de entrada existente, asumiendo que la resistencia de la fuente tienda a cero se tendría el circuito que se muestra en la figura 1.18.

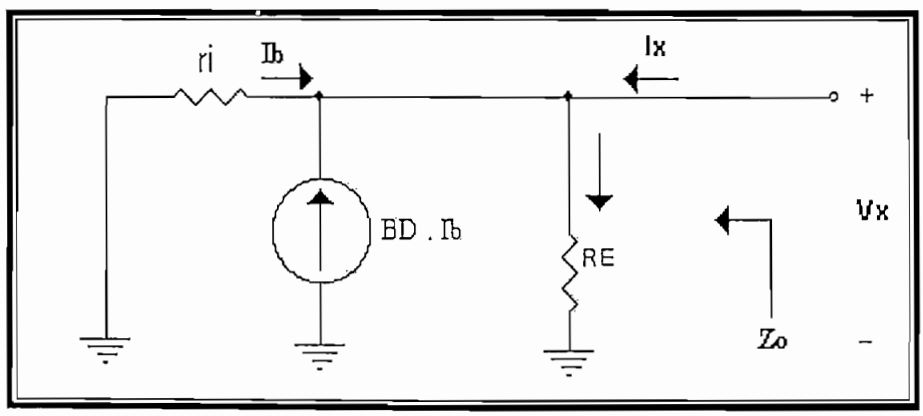


FIGURA 1.18) Circuito AC equivalente con un voltaje V_x a la salida y cortocircuitando a la fuente voltaje de la figura 1.16

La corriente I_x generada por la fuente V_x es igual:

$$I_x = \frac{V_x}{R_E} + \frac{V_x}{r_i} - \beta_D . I_b = \frac{V_x}{R_E} + \frac{V_x}{r_i} - \beta_D . \left(\frac{-V_x}{r_i} \right) \qquad \text{Ec. 1.37}$$

$$I_x = \left(\frac{1}{R_E} + \frac{1}{r_i} + \frac{\beta_D}{r_i} \right) . V_o \qquad \text{Ec. 138}$$

Resolviendo para Z_o se tiene:

$$Z_o = \frac{V_x}{I_x} = \frac{1}{1/R_E + 1/r_i + \beta_D / r_i} = R_E \,||\, r_i \,||\, \frac{r_i}{\beta_D} \qquad \text{Ec.1.39}$$

$$Z_o \approx \frac{r_i}{\beta_D} \qquad \text{Ec. 1.40}$$

En conclusión, un circuito Darlington tiene una alta impedancia de entrada y una baja impedancia de salida. Además este circuito ofrece una gran amplificación de corriente y una amplificación de voltaje que tiende a 1.

1.5.1.2 Par Retroalimentado

Un par retroalimentado es un circuito conformado por dos transistores que operan en forma similar a un circuito Darlington. Este par está conformado por un transistor *PNP* que excita a un transistor NPN, que en conjunto actúan como un transistor *PNP*. (Véase Figura 1.19)

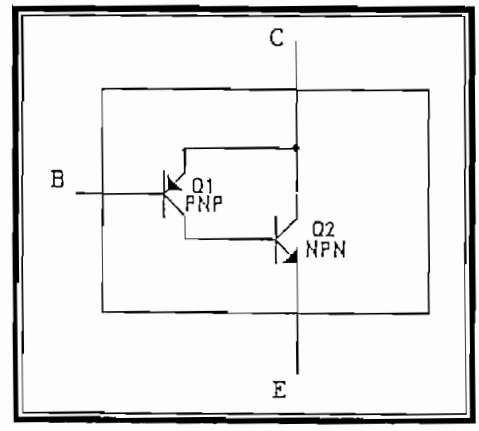


FIGURA 1.19) Circuito Par retroalimentado

Un circuito que muestra la polarización de un Par Retroalimentado se lo ha representado en la figura 1.20 y su respectivo equivalente en AC, se lo demuestra en las figuras 1.21 y 1.22.

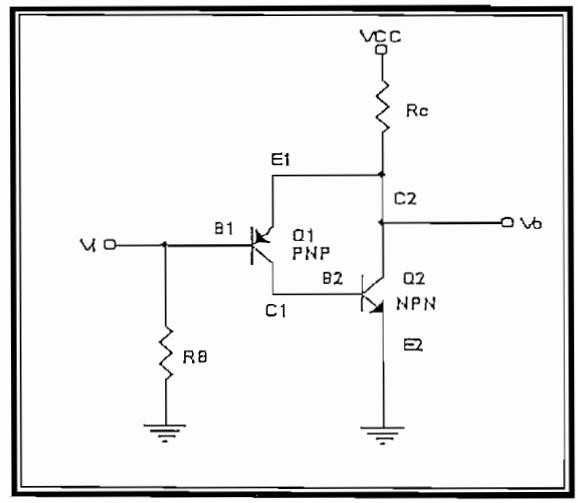


FIGURA 1.20) Circuito amplificador con par retroalimentado

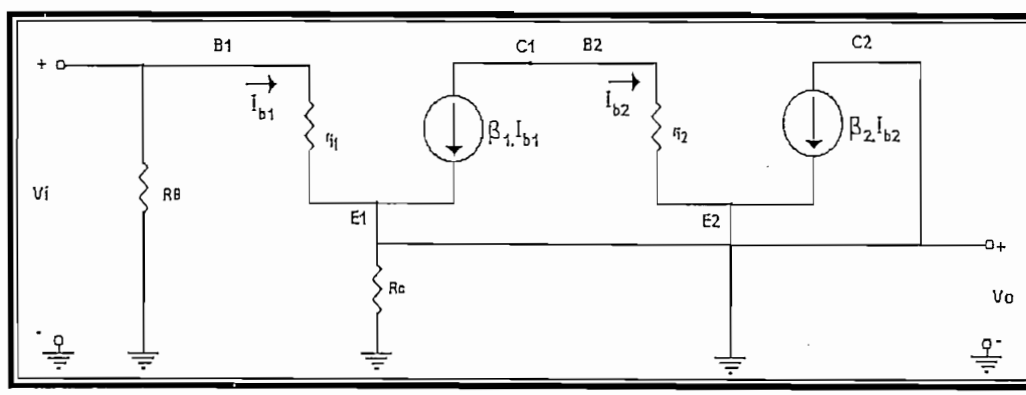


FIGURA 1.21) Circuito AC equivalente 1 del circuito de la figura 1.20

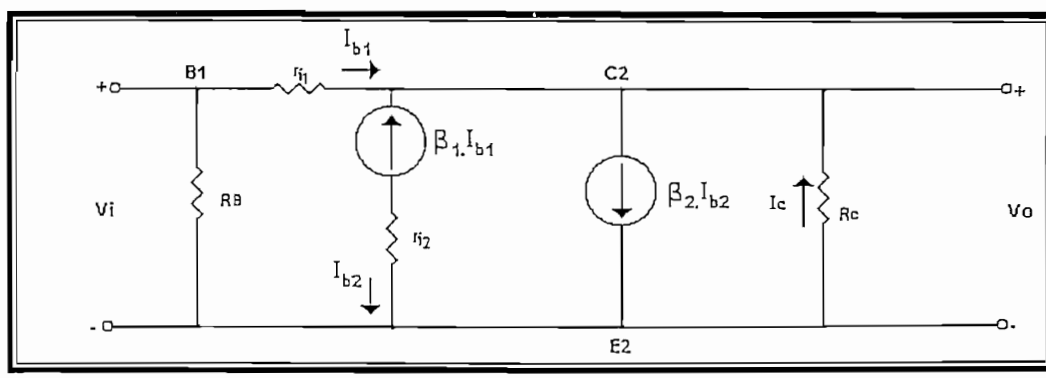


FIGURA 1.22) Circuito AC equivalente 2 del circuito de la figura 1.20

- **Impedancia de Entrada (Z_I)**

La impedancia de entrada vista desde la fuente de señal de acuerdo a la figura 1.21 es igual a:

$$Z_{In} = R_B \,||\, \frac{V_I}{I_{b1}} \qquad \text{Ec. 1.41}$$

En donde:

$$I_{b1} = \frac{V_i - V_o}{r_{i1}} \qquad \text{Ec.1.42}$$

Tomando en cuenta que $I_{b2} = I_{C1} = \beta_1 . I_{b1}$, el voltaje de salida V_O puede expresarse de la siguiente manera:

$$V_o = -I_{RC} . R_C \approx \left(-\beta_1 . I_{b1} - I_{b1} + \beta_2 . I_{b2}\right) . R_C \approx \left(\beta_2 . I_{b2}\right) . R_C \qquad \text{Ec. 1.43}$$

Entonces se obtiene:

$$I_{b1} r_{i1} = V_i - V_o \approx V_i - \left(\beta_2 . I_{b2}\right) . R_C \qquad \text{Ec. 1.44}$$

$$I_{b1} . r_{i1} + \beta_2 . \left(\beta_1 . I_{b1}\right) . R_C = V_i \qquad \text{Ec. 1.45}$$

$$\frac{V_i}{I_{b1}} = r_{i1} + \beta_1 . \beta_2 . R_C \qquad \text{Ec. 1.46}$$

La impedancia de entrada es igual a:

$$Z_I = R_B \,||\, \left(r_{i1} + \beta_1 . \beta_2 . R_C\right) \qquad \text{Ec. 1.47}$$

- **Impedancia de Salida (Z_O)**

Para determinar la impedancia de salida, de igual manera que en los análisis anteriores, se cortocircuita la fuente de entrada y se conecta a la salida una fuente de voltaje V_x y corriente I_x como se demuestra en la figura 1.23.

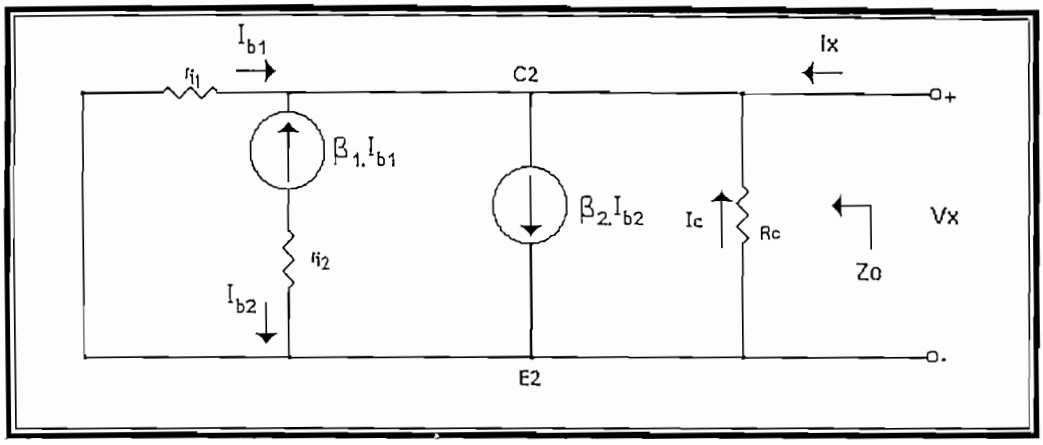


FIGURA 1.23) Circuito AC equivalente con un voltaje V_x a la salida y cortocircuitando a la fuente voltaje de la figura 1.20

De acuerdo a la figura 1.23 se tiene que Z_o es igual:

$$Z_o = \frac{V_x}{I_x} \qquad \text{Ec. 1.48}$$

Aplicando la Ley de corriente, se tiene:

$$I_x = \left(I_{b2} + \beta_2 . I_{b2}\right) - \beta_1 . I_{b1} - I_{b1} - I_C \qquad \text{Ec. 1.49}$$

$$I_x = \beta_2 (\beta 1 . Ib1) - \beta_1 . I_{b1} - I_{b1} - I_C \qquad \text{Ec. 1.50}$$

$$I_x = \frac{\beta_1 . \beta_2 . V_x}{r_{i1}} - \frac{-\beta_1 . V_x}{r_{i1}} - \frac{-V_x}{r_{i1}} - \frac{-V_x}{R_c} \qquad \text{Ec. 1.51}$$

Despejando V_x y pasando al primer lado de la ecuación:

$$\frac{I_x}{V_x} = \frac{\beta_1 . \beta_2}{r_{i1}} + \frac{\beta_1}{r_{i1}} + \frac{1}{r_{i1}} + \frac{1}{R_c} \qquad \text{Ec. 1.52}$$

Por último se tiene que Z_o es igual:

$$Z_o = R_c \,||\, r_{i1} \,||\, \frac{r_{i1}}{\beta_1} \,||\, \frac{r_{i1}}{\beta_1 . \beta_2} \qquad \text{Ec. 1.53}$$

$$Z_o \approx \frac{r_{i1}}{\beta_1 . \beta_2} \qquad \text{Ec. 1.54}$$

- **Ganancia de Corriente (A_i)**

La ganancia de corriente para este circuito según la figura 1.22 está dada por:

$$A_i = \frac{I_o}{I_i} = \frac{I_o}{I_{b1}} \frac{I_{b1}}{I_i} \qquad \text{Ec. 1.55}$$

La corriente de salida I_o es igual:

$$I_o = \beta_2 . I_{b2} - \beta_1 . I_{b1} - I_{b1} \qquad \text{Ec. 1.56}$$

$$I_o = \beta_2 (\beta_1 . I_{b1}) - (1 + \beta_1) Ib1 \approx \beta_2 . \beta_1 . I_{b1} \qquad \text{Ec. 1.57}$$

$$\frac{I_o}{I_{b1}} = \beta_1 . \beta_2 \qquad \text{Ec. 1.58}$$

Para expresar la relación I_{b1}/I_i se usa la regla del divisor de corriente:

$$I_{b1} = \frac{R_B}{Z_i + R_B} . I_i \qquad \text{Ec. 1.59}$$

La ganancia de corriente del circuito queda definida como:

$$A_i = \beta_1 . \beta_2 . \frac{R_B}{R_B + Z_i} \qquad \text{Ec. 1.60}$$

- **Ganancia de Voltaje (A_V)**

La ganancia de voltaje para el circuito 1.22 se define como:

$$A_V = \frac{V_o}{V_i} \qquad \text{Ec. 1.61}$$

El voltaje de salida es:

$$V_o = -I_{RC}.R_C \approx \beta_2.\beta_1.I_{b1}.R_C \qquad \text{Ec. 1.62}$$

Debido a que:

$$I_{b1} = \frac{V_i - V_o}{r_{i1}} \qquad \text{Ec. 1.63}$$

Se tiene:

$$V_o = V_i - I_{b1}.r_{i1} = V_i - \frac{V_o}{\beta_1.\beta_2.R_c}.r_{i1} \qquad \text{Ec. 1.64}$$

La ganancia de voltaje queda definida así:

$$A_V = \frac{1}{1 + r_{i1}/(\beta_1.\beta_2.R_c)} = \frac{\beta_1.\beta_2.R_c}{\beta_1.\beta_2.R_c + r_{i1}} \approx 1 \qquad \text{Ec. 1.65}$$

El par retroalimentado proporciona una ganancia en corriente muy alta y una ganancia de voltaje que tiende a 1 tal como sucede con un Par Darlington, de igual manera proporciona una baja impedancia a la salida y una alta impedancia de entrada.

1.6 CIRCUITOS INTEGRADOS

Los circuitos amplificadores de audio antes mencionados, en la actualidad se los puede encontrar en pequeños circuitos integrados, ayudando a reducir espacio, complejidad del circuito, entre otras ventajas; sin embargo tienen algunas desventajas que se las menciona a continuación:

- La información que proporcionan los fabricantes en algunos casos no es suficiente para poder desarrollar un diseño personal sobre la base del CI, por ejemplo el STK 4141.

- La utilización de un CI en un circuito proporciona mayor facilidad en la construcción y reparación si existiese alguna avería (menos elementos), sin embargo si el problema es el CI, sería muy costosa la reparación.

- Otro factor que se da en nuestro medio, es que no se pueden encontrar los CI originales y si los hay son muy costosos, por tal razón en el mercado se encuentran los clones de los CI, más baratos, pero algunos con un funcionamiento deficiente.

Afortunadamente estas desventajas no son para todos los CI, pero para el caso de los CI amplificadores de audio se han presentado estos inconvenientes.

En esta oportunidad se va a explicar la estructura interna y el funcionamiento de un CI amplificador de audio, pero para reforzar estas explicaciones se va a realizar un diseño de un amplificador de audio con elementos discretos (Transistores Bipolar de Juntura).

1.6.1 ESTRUCTURA INTERNA DE UN CIRCUITO INTEGRADO AMPLIFICADOR DE AUDIO

La estructura interna de un circuito integrado amplificador de audio, tiene como base ciertas estructuras electrónicas, que combinadas entre sí, trabajan para la realización de esta operación. Estas combinaciones dependen de la casa fabricante o de la potencia de salida que se quiera obtener.

Los circuitos que comúnmente se encuentran en los diagramas internos de estos circuitos integrados son:

- Par Darlington
- Par retroalimentados
- Fuentes de corriente
- Espejos de corriente
- Amplificadores Diferenciales

Los circuitos Par Darlington y Par retroalimentados ya se los describió anteriormente (véase 1.5.1.1 y 1.5.1.2), por lo que brevemente se detallará los circuitos restantes.

1.6.1.1 Fuentes de Corriente

Una fuente de corriente es la encargada de suministrar una corriente constante sin importar la carga que esté conectada a ella; una de las características más importantes de las fuentes de corriente, es la de presentar una impedancia muy alta.

A continuación se diagrama una fuente de corriente constante a base de un transistor bipolar (Figura 1.24).

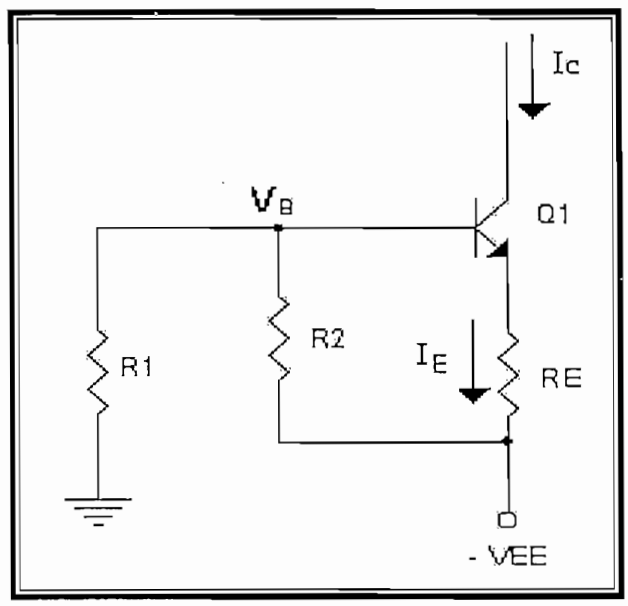


FIGURA 1.24) Circuito fuente de corriente

La figura 1.24 señala un circuito que utiliza unas cuantas resistencias y un transistor NPN para operar como un circuito de corriente constante. La corriente I_C es la proporcionada por el circuito en forma constante, que depende de los valores de las resistencias R_1 y R_2 como se describe en las siguientes expresiones:

$$V_B = \frac{R_1}{R_1 + R_2}.(-V_{EE}) \qquad \text{Ec. 1.66}$$

La expresión anterior es posible siempre y cuando la impedancia de entrada de la base sea mucho más grande que R_1 y R_2. El voltaje en emisor será:

$$V_E = V_B - V_{BE} = V_B - 0.7 \qquad \text{Ec. 1.67}$$

Con lo que se obtiene que:

$$I_E = \frac{V_E - (-V_{EE})}{R_E} \approx I_C \qquad \text{Ec. 1.68}$$

Si se reemplaza R_2 (Figura 1.24) con un diodo zener, como se muestra en la figura 1.25, se obtiene una mejor fuente de corriente constante, debido a que el diodo proporciona un voltaje constante a la base del transistor con lo que se obtiene que la corriente controlada no dependa del voltaje de alimentación V_{EE}.

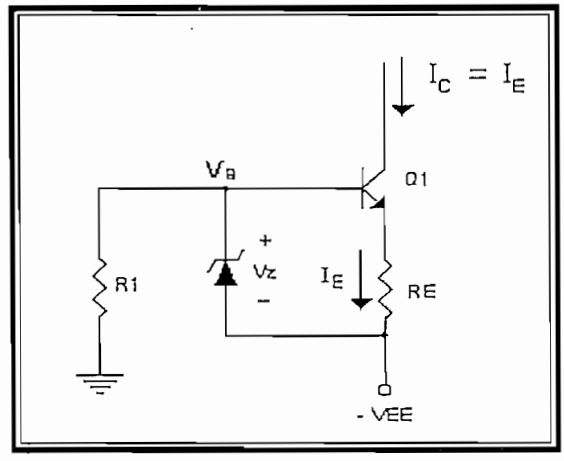


FIGURA 1.25) Fuente de corriente con un diodo zener

1.6.1.2 Espejos de Corriente

En los CI's se acostumbra a mantener corrientes (I) constantes para los TBJ´s que efectúan la función amplificadora. Esto se logra utilizando TBJ's que actúen como fuentes de corriente constante de valor (I). Además en la fabricación de CI's se requiere que los transistores utilizados tengan idénticas caídas de voltaje base-emisor e idénticos valores de beta.

Un circuito espejo de corriente, es un circuito formado por un par de transistores, arreglados de tal manera, para que por los mismos circule una corriente idéntica ($I_X = I$); es decir, por un transistor ingresa una corriente la cual se refleja como la corriente de entrada del otro transistor, como se grafica en la figura 1.26.

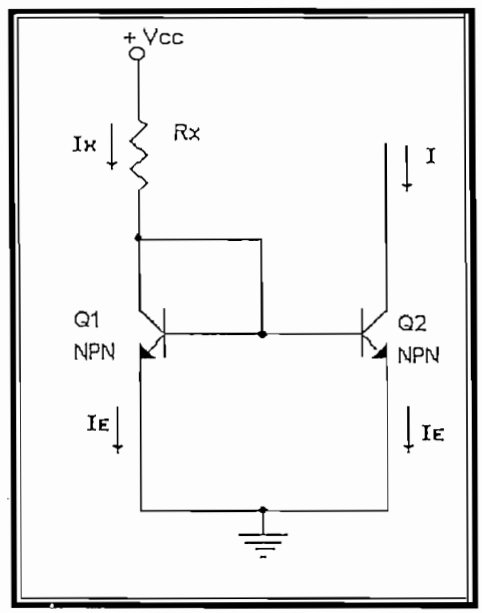


FIGURA 1.26) Circuito espejo de corriente

Para deducir que $I_X = I$, se supone que la corriente de emisor (I_E) para ambos transistores es la misma (siendo Q1 y Q2 fabricados uno junto al otro en el mismo microcircuito). El sentido de las corrientes se muestra en la figura 1.27.

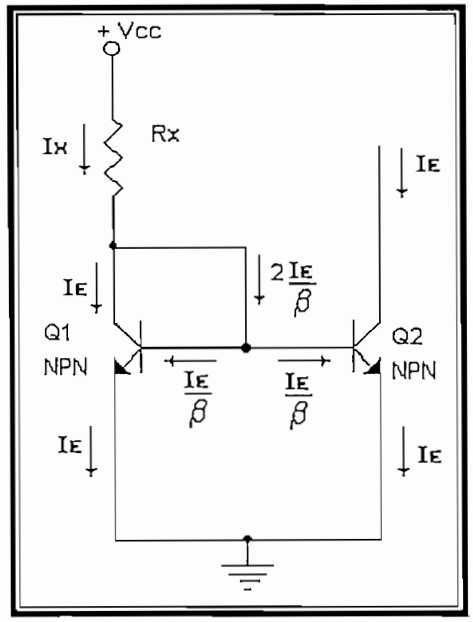


FIGURA 1.27) Flujo de corrientes en un circuito espejo de corriente

Las corrientes de base de cada transistor son:

$$I_B = \frac{I_E}{\beta + 1} \approx \frac{I_E}{\beta} \qquad \text{Ec. 1.69}$$

Asumiendo que $I_C \approx I_E$.

La corriente a través del resistor R_X , I_X es la siguiente:

$$I_X = I_E + \frac{2.I_E}{\beta} = \frac{\beta.I_E}{\beta} + \frac{2.I_E}{\beta} = \frac{\beta + 2}{\beta} I_E \approx I_E \qquad \text{Ec. 1.70}$$

Con lo que se tiene como resultante que $I_X \approx I$, debido a que $I_X \approx I$, es decir la corriente I_X que depende de V_{CC} y R_X se refleja en la corriente (I) del colector de Q_2.

1.6.1.3 Amplificadores Diferenciales

Un circuito amplificador diferencial es una conexión muy popular y se utiliza en circuitos integrados, este circuito se muestra en la figura 1.28.

Este es un circuito con dos entradas separadas y dos salidas separadas, existiendo una conexión entre los transistores por medio de los emisores. Existe la posibilidad de tres combinaciones de la señal de entrada, las cuales son:

- Si se aplica una señal a alguna entrada, estando la otra entrada conectada a tierra se tiene una operación llamada de *una sola terminal.* En esta operación debido a la conexión de emisor común, la señal de entrada opera en ambos transistores dando como resultado una salida en ambos colectores.

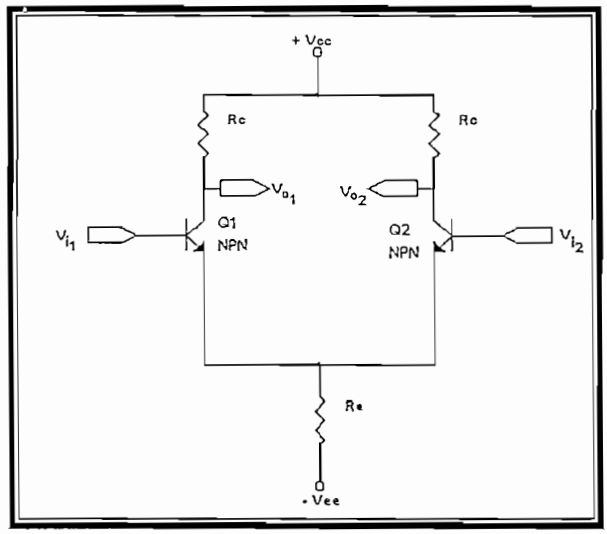


FIGURA 1.28) Circuito amplificador diferencial

- Si se aplican dos señales de entrada de polaridad opuesta, la operación se llama de *doble terminal.* En este tipo de operación se tiene salidas en ambos colectores, estas señales que se encontran a las salidas son la amplificación de la diferencia de las señales aplicadas en las entradas.

- Si se aplica la misma señal a ambas entradas la operación se llama de *modo común*. Esta entrada común da como resultado señales opuestas en cada colector, la señal de salida resultante es muy pequeña y tiende a cero debido a que se anulan las señales de salida de cada *transistor.*

El circuito equivalente AC del amplificador diferencial es el siguiente:

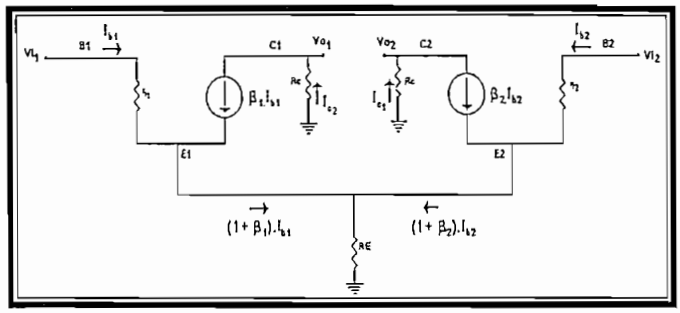


FIGURA 1.29) Circuito AC equivalente del circuito de la figura 1.28

En donde se supone que:

$$I_{b1} = I_{b2} = I_b$$

$$r_{i1} = r_{i2} = r_i$$

$$\beta_1 = \beta_2 = \beta$$

En resumen se puede tener el siguiente cuadro:

MODO DE OPERACIÓN	ENTRADAS	SALIDAS	GANANCIA
Un terminal	Se aplica una señal a cualquiera de las entradas estando la otra conectada a tierra.	Una misma señal de salida en ambos colectores.	$A_v = \frac{V_o}{V_{i1}} = \frac{R_C}{2.r_e}$
Doble terminal	Se aplican señales de entrada diferentes	Se tiene en cada colector la amplificación de la diferencia de las señales de entradas.	$A_d = \frac{V_o}{V_{i1} - V_{i2}} = \frac{R_C}{2.r_e}$
Modo común	Se aplica la misma señal a ambas entradas.	Señales opuestas en cada colector, de similar magnitud. Entre colectores se tiene aproximadamente cero voltios	$A_C = \frac{V_o}{V_i} = \frac{\beta.R_C}{r_i + 2.(\beta + 1)R_E}$

TABLA 1.2) Modos de operación de amplificadores diferenciales

Los amplificadores diferenciales tienen una gran aceptación, por la mínima distorsión que generan, para poder cuantificar este parámetro existe una relación llamada de *rechazo en modo común*, con la cual se determina si un amplificador es mejor que otro. Este rechazo en modo común se refiere a la relación entre la ganancia en modo de doble terminal y la ganancia en modo común de un amplificador diferencial.

La justificación de esta relación se da debido a que una característica importante de una conexión de doble terminal diferencial, es que las señales que son opuestas en las entradas tienen una amplificación muy alta, mientras que las que son comunes a las dos entradas son ligeramente amplificadas; la operación global amplifica la señal diferencial, mientras que rechaza (amplifica ligeramente) la señal común a las dos entradas.

Esto ayuda a eliminar la distorsión debido a que el ruido o cualquier señal de entrada no deseada es por lo general común a ambas entradas, la conexión de doble terminal diferencial tiende a proporcionar atenuación de esta entrada no deseada.

Para que la operación de un amplificador diferencial mejore, la ganancia en modo común del circuito tiene que ser lo más pequeña posible, idealmente que tienda a cero (relación *rechazo en modo común* lo más grande posible), esto se logra teniendo una resistencia R_E que tienda al infinito. Para obtener esto se coloca una fuente de corriente constante en reemplazo de la resistencia R_E, como se muestra en la figura 1.30.

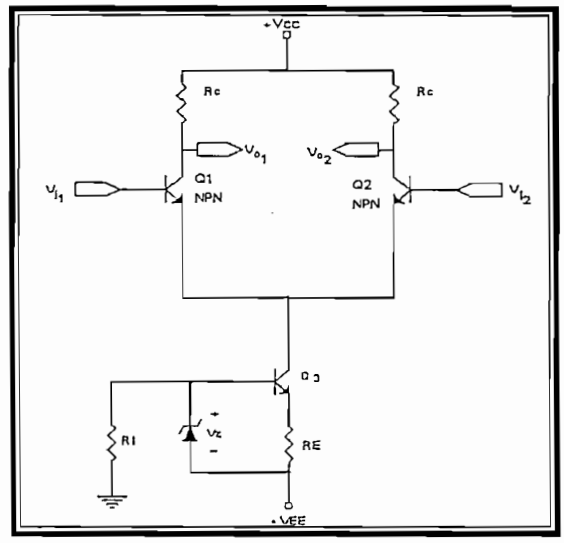


FIGURA 1.30) Amplificador diferencial con fuente de corriente conectada a los emisores para aumentar la relación de rechazo en modo común.

1.6.1.4 STK 4141

El STK 4141 es un amplificador de audio de dos canales que entrega 25 W por canal con una Distorsión Armónica Total (TDH) máxima igual al 0.3 %. La estructuración interna de este integrado se presenta en la figura 1.31, la cual muestra los circuitos antes citados que combinados trabajan como un amplificador de audio.

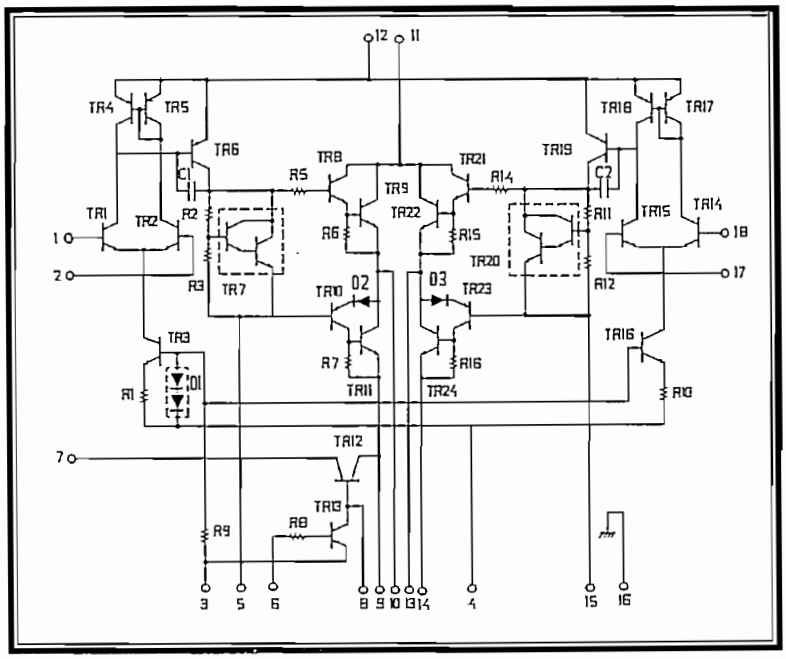


FIGURA 1.31) Diagrama interno del CI STK4141

La figura anterior muestra el diagrama interno del circuito integrado STK4141. En esta estructura se pueden identificar las etapas que conforman un amplificador de audio que anteriormente se mencionó, la identificación se muestra en la figura 1.32.

La figura 1.32 muestra el circuito interno correspondiente a un canal del STK4141, en este se determinan cada una de las etapas, cuyo funcionamiento ya se explicó anteriormente. La etapa de retroalimentación no consta debido a que ésta se forma con elementos externos.

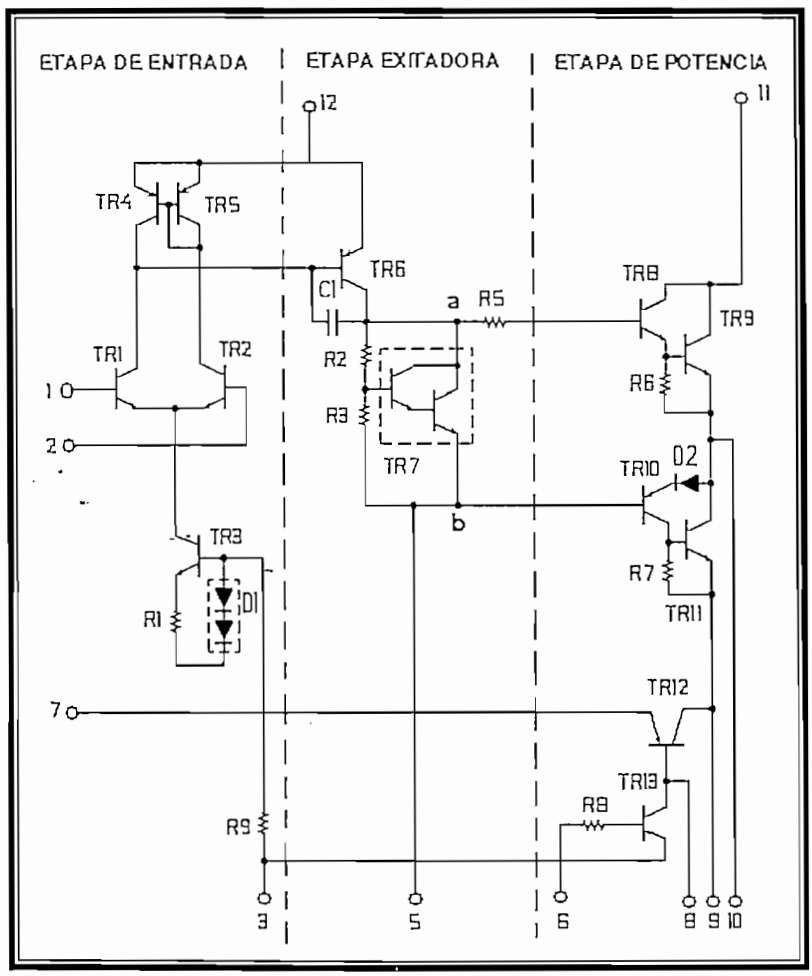


FIGURA 1.32) Etapas que conforman un amplificador de audio

A continuación se describirá brevemente el trabajo que realiza las diferentes partes del circuito interno del STK 4141, teniendo como referencia la figura 1.33.

- Estructura 1

La estructura 1 está conformada por un circuito espejo de corriente, un amplificador diferencial y una fuente de voltaje. El circuito espejo de corriente esta conectado a los colectores del circuito amplificador diferencial con la finalidad de proporcionar corrientes de colector totalmente iguales de tal manera que las corrientes y voltajes en los transistores T_{R1} y T_{R2} sean iguales, con esto se obtiene que la distorsión sea mínima.

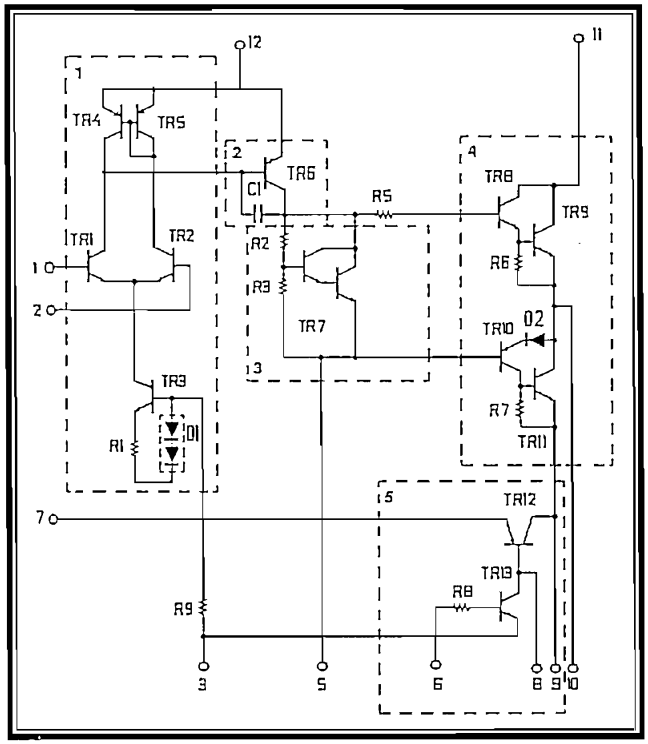


FIGURA 1.33) Estructuras que conforman el STK4141

La fuente de corriente va conectada a los emisores del amplificador operacional con lo que se logra que la resistencia de emisor tienda a infinito logrando reducir la ganancia en modo común lo que a su vez influye en el aumento de la relación de rechazo en modo común (CMRR), que como ya se vio anteriormente mientras más grande es el valor de CMRR menos susceptible es el amplificador diferencial a señales de ruido, teniéndose poca distorsión.

La operación global de ésta etapa es recoger la señal de entrada ya sea por el terminal positivo (T_{R1}) o por el terminal negativo (T_{R2}) y entregar por uno de sus colectores una señal amplificada de la diferencia de la señal de entrada con la señal proveniente de la red de realimentación[3]. La principal característica de esta etapa es realizar todo este proceso con la menor distorsión posible, debe tomarse en cuenta que la señal que se obtiene del colector tiene el sentido contrario de la señal de entrada.

Las resistencias R_1 y R_9 así como el diodo D_1, que forman parte de la fuente de corriente se encargan polarizar a T_{R3} y fijar el valor de corriente que entrega la fuente.

- **Estructura 2**

Como se menciono en la etapa 1, la salida del colector del amplificador diferencial es inversa a la señal de entrada con un nivel bajo de voltaje, por lo que es necesario invertir nuevamente esta señal y amplificar a un valor adecuado de voltaje para obtener el nivel de potencia esperado.

Esta inversión de la señal se logra con un transistor (T_{R6}), el cual recibe la señal por la base y entrega una señal invertida y amplificada por el colector para que pueda ser procesada por las etapas posteriores.

Cabe hacer notar que la estructura tal como se muestra no tiene un buen grado de amplificación, por lo que se hace indispensable conectar dispositivos periféricos (resistencias) que fijen el valor de la ganancia de voltaje que se necesitará para poder proporcionar la potencia requerida, ya que en las etapas posteriores se amplifica únicamente el nivel de corriente.

---

[3] La red de realimentación no consta en la figura 1.33, ya que esta se forma en la parte exterior al CI STK4141.

El capacitor C_1 trabaja como un circuito realimentador, utilizado para eliminar cualquier tendencia a la inestabilidad para altas frecuencias.

- **Estructura 3**

Esta estructura está conformada por un circuito Darlington (T_{R7}) y dos resistencias (R_2 y R_3) que se utilizan para polarizar al T_{R7}. La función de esta etapa es entregar a la base del transistor T_{R10} una señal semejante en forma a la señal que ingresa por la base del transistor T_{R8} pero defasada 180°.

La resistencia R_5 se utiliza para limitar la señal que ingresa al transistor T_{R8}, de tal manera que sea igual en magnitud a la señal que ingresa al T_{R10}, de esta forma se logra que no exista una descompensación en las entradas de T_{R8} y T_{R10} y no haya problemas de fallas térmicas o distorsión.

Como ya se menciono las resistencias R_2 y R_3 polarizan al T_{R7} para que este entregue a las bases de T_{R8} y T_{R9} corrientes adecuadas que polaricen a estos, para que no haya distorsión por cruce por cero. Esto se puede lograr haciendo que I_{CQ} de T_{R8} y T_{R9} este ligeramente por encima de cero.

- **Estructura 4**

Esta estructura está conformada por un par Darlington y un par retroalimentado, cuya función es elevar el nivel de corriente de la señal que proviene de las estructuras anteriores. La forma de operación de esta estructura es igual al de un circuito clase B de simetría complementaria que ya se lo vio anteriormente.

El par Darlington está formado por T_{R8} y T_{R9}, este par trabaja solo en el semiciclo positivo de la señal proveniente del colector del T_{R7}, es decir reproduce en el emisor de T_{R9} el semiciclo positivo de la señal que ingresó a la base de T_{R8}, con niveles de corriente mayores.

La resistencia R_6 se encarga de limitar la corriente que ingresa a la base del transistor T_{R9} y además mejora la impeancia de entrada del circuito Darlington.

El par retroalimentado está formado por T_{R10} y T_{R11}, este trabaja solamente en el semiciclo negativo de la señal que proviene del emisor de T_{R7}, al igual que el par Darlington, amplifica el nivel de corriente.

La resistencia R_7 cumple la misma función que la resistencia R_6 pero para la base de T_{R11}. El diodo D_2 es utilizado para fijar el valor del voltaje de base-emisor de este par retroalimentado en 1.3 Voltios para que sea igual al voltaje base-emisor del par Darlington y no haya descompensación en las entradas.

- Estructura 5

La estructura 5, esta constituida por dos transistores (T_{R12} y T_{R13}) interconectados entre sí y que trabajan en la región de corte o saturación según el valor de voltaje que ingrese por el terminal *6* (0 ó –5 voltios) que va a la entrada de la base de T_{R13}.

La salida total de esta estructura está en el emisor de T_{R12} (terminal 7), en la cual se puede obtener $-V_{CC}$ ó 0 según se la entrada 0 ó –5 respectivamente. Estas salidas de voltaje sirven para polarizar o no a las estructuras antes mencionadas provocando que el CI funcione o deje de funcionar. A la operación de este circuito de le conoce como "muting".

A continuación se presenta el circuito de la estructura 5 visto de otra forma:

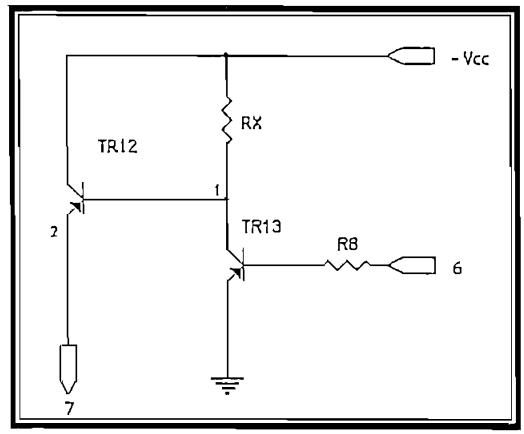


FIGURA 1.34) Estructura 6 vista de otra manera

Tomándose en cuenta que para qué un transistor PNP entre en corte o en saturación deben darse lo siguientes voltajes de polarización:

CORTE:	$V_C \leq V_B$	y	$V_E \leq V_B$	(TBJ = circuito abierto)
SATURACIÓN:	$V_B \leq V_C$	y	$V_B \leq V_E$	(TBJ = corto circuito)

Según esto se tendría la siguiente tabla:

Terminal 6	Punto 1	Estado TR13	Punto 2	Estado TR12	Salida 4
0 V	$- V_{cc}$	Abierto	$- V_{cc}$	cerrado	$- V_{cc}$
- 5	0 V	Cerrado	0 V	abierto	0 V

TABLA 1.3) Características del circuito de Muting

La resistencia R_X se la dimensiona de tal forma que polarice al T_{R13} para que trabaje en saturación.

Para evitar que cualquier señal parásita que haya ingresado a este circuito afecte al voltaje DC que se obtiene en la salida (terminal 7), se coloca un capacitor de un valor pequeño dividiendo a la resistencia de colector R_X en dos, de tal manera que si existe alguna señal de este tipo se dirija a tierra y no a la salida. El circuito que describe lo antes mencionado es el siguiente:

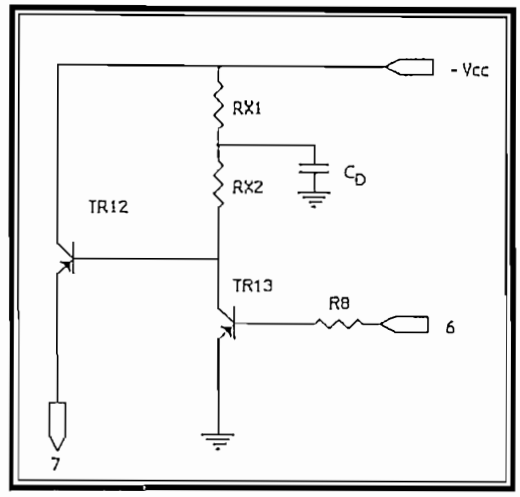


FIGURA 1.35) Estructura 6 con un capacitor de desvío para evitar señales parásitas

Un ejemplo de la utilización del STK 4141 es un amplificador de audio el cual se muestra en la figura 1.36:

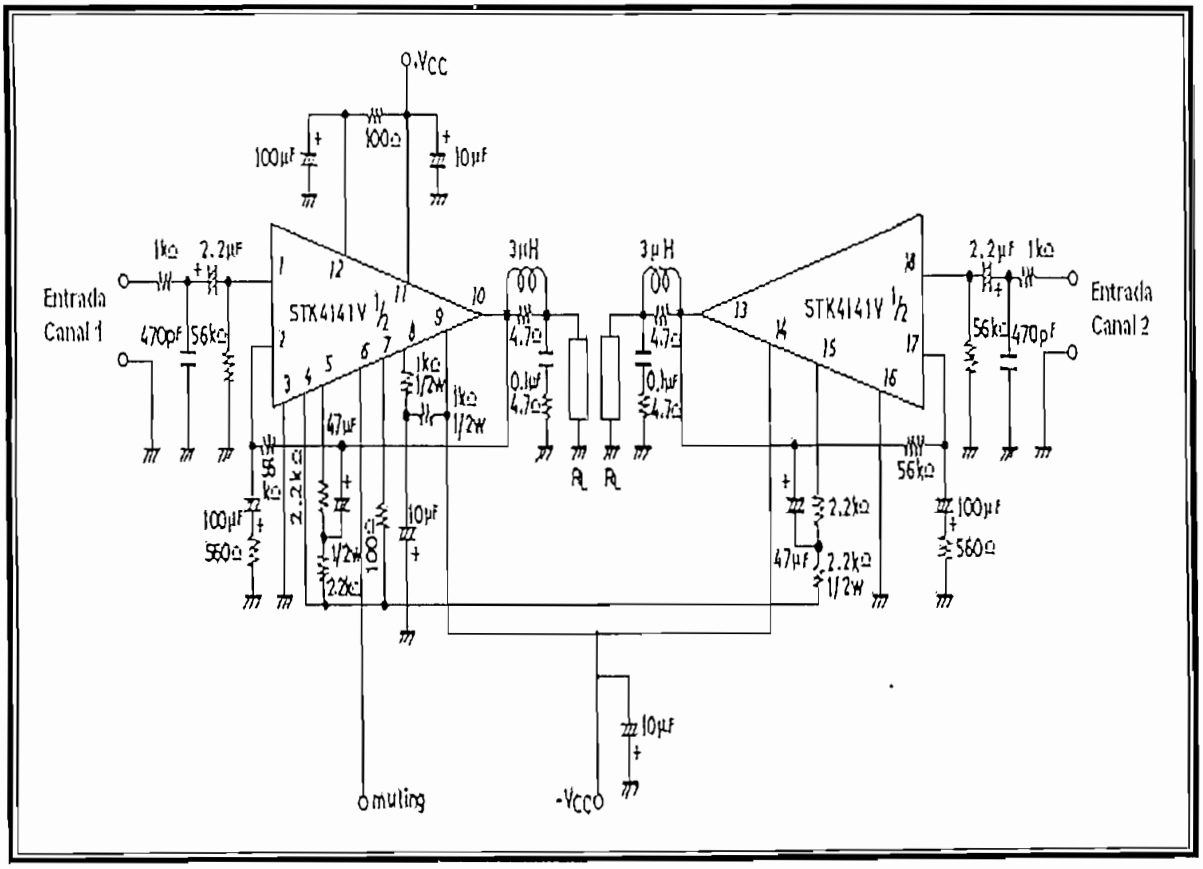


FIGURA 1.36) Circuito de un amplificador de audio estereo de 25 Vatios basado en el C.I. STK 4141

1.7 MODOS DE CONEXIÓN A LA SALIDA

En los amplificadores o etapas de potencia dedicados a la amplificación de audio, es importante dar un acoplamiento adecuado entre la salida y la carga, la misma que consiste en que la resistencia de salida sea mucho menor que la resistencia de la carga.

Se debe considerar que la impedancia nominal de los altavoces puede ser de 2Ω, 4Ω, 6Ω, 8Ω, o 16Ω (generalmente 8Ω), la resistencia de salida de la etapa de potencia tiene que ser muy pequeña.

Para que exista un adecuado acoplamiento entre estas impedancias, se han adoptado dos tipos de conexiones o modos de funcionamiento, como se detallan a continuación.

MODO ESTÉREO

Este modo se caracteriza por tener un doble proceso para una misma fuente, tal es así que, en cada canal se realiza la respectiva amplificación de la señal de entrada independientemente del otro canal, y adicionalmente mantiene sus salidas independientes.

Entre otras ventajas de este modo, es la de obtener la regulación de la ganancia de cada canal sin que esto afecte al otro. Este modo de conexión es el más general en el ámbito profesional.

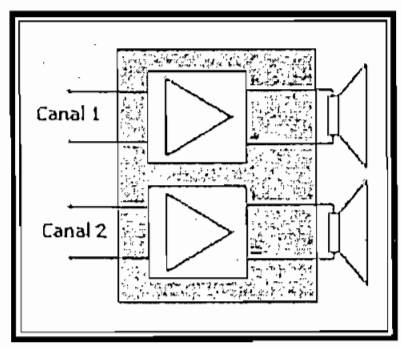


FIGURA 1.37) Diagrama de bloques de un circuito en modo estereo

En la figura anterior, se muestra una etapa o amplificador de potencia estéreo. Cada rectángulo representa un amplificador de potencia. Según la configuración del aparato pueden compartir o no la fuente de alimentación.

1.7.1 MODO PARALELO O MONO

A menudo se requiere que los dos canales lleven la misma señal, sin independencia entre sí. Esto se consigue poniendo las entradas de los canales en paralelo, conectadas a la señal ya amplificada.

Para tener mayor versatilidad en los amplificadores, éstos habitualmente deben constar de un conmutador que activa el modo paralelo, el mismo que simplemente pone las dos entradas del amplificador en paralelo[4] para que lleven la misma señal de entrada.

Normalmente, para seleccionar el modo paralelo/mono, el amplificador toma la señal del canal 1 y se desconecta la entrada del canal 2. Dependiendo del modelo de amplificador, cada canal seguirá manteniendo su control de ganancia, o bien el canal 1 controlará el nivel de señal que va a los amplificadores de ambos canales.

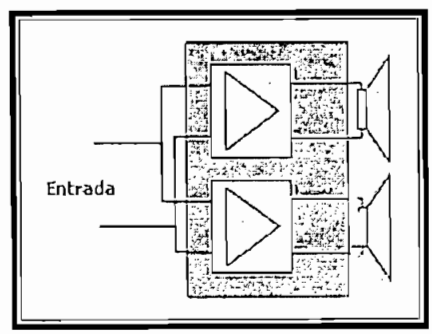


FIGURA 1.38) Diagrama de bloques de un circuito en modo paralelo o mono

[4] Solo se debe conectar la señal de entrada a un canal del amplificador

1.8 CARACTERÍSTICAS PRINCIPALES DE LOS AMPLIFICADORES COMERCIALES

Existen algunas características de los amplificadores que nos ayudan a entender las especificaciones técnicas dadas por el fabricante que identifican su funcionamiento. Entre las cuales se tiene:

1.8.1 POTENCIA DE SALIDA

Es la máxima potencia que se entrega a la carga, clasificándose ésta en dos tipos según los fabricantes:

1.8.1.1 Potencia Nominal

Conocida también como potencia RMS o eficaz, se define como la potencia que el amplificador es capaz de proporcionar a la carga nominal tomando como señal de entrada para los dos canales un tono senoidal puro de 1 KHz, la máxima potencia nominal que puede entregar este amplificador se da hasta que el nivel de distorsión armónica total (THD) no sobrepase el valor que da el fabricante, entonces se dice que se ha alcanzado la potencia nominal.

Debido a que la señal musical que suele excitar los amplificadores, se parece muy poco a la señal sinusoidal usada para medir la Potencia Nominal, se recurre a la Potencia Musical.

1.8.1.2 Potencia Musical o Dinámica

Las señales de audio no tienen una forma de onda especifica o que se la pueda simular de alguna manera, por lo que algunos han adoptado como señal de prueba una señal senoidal de 1 KHz en la cual cada 20 ms hay un pico cuyo nivel pasa a ser diez veces mayor, de esta manera se puede cuantificar el nivel de potencia máximo a intervalos cortos de tiempo.

Por este motivo, a la hora de decidir entre dos amplificadores, es mejor contar con la información de la potencia nominal.

En las especificaciones técnicas de una etapa o amplificador de potencia, se habla simplemente de "potencia", esta se refiere usualmente a la potencia nominal o eficaz. Los sufijos (musical o dinámica) se utilizan más en amplificadores domésticos, a veces, para "aumentar" la potencia real del aparato y otras para complicar su comprensión.

1.8.2 RESPUESTA EN FRECUENCIA.

Se llama respuesta en frecuencia al comportamiento de un dispositivo de audio frente a las distintas frecuencias que componen el espectro de audio (20 a 20.000 Hz). La razón de analizar la respuesta de estos dispositivos en el dominio de la frecuencia se da por el hecho de que cada uno de estos dispositivos (micrófonos, amplificadores, parlantes) aunque trabajan en el mismo espectro de audio no reaccionan igual a todas las frecuencias.

A continuación se indica un ejemplo práctico en el cual se muestra la respuesta en frecuencia de un altavoz.

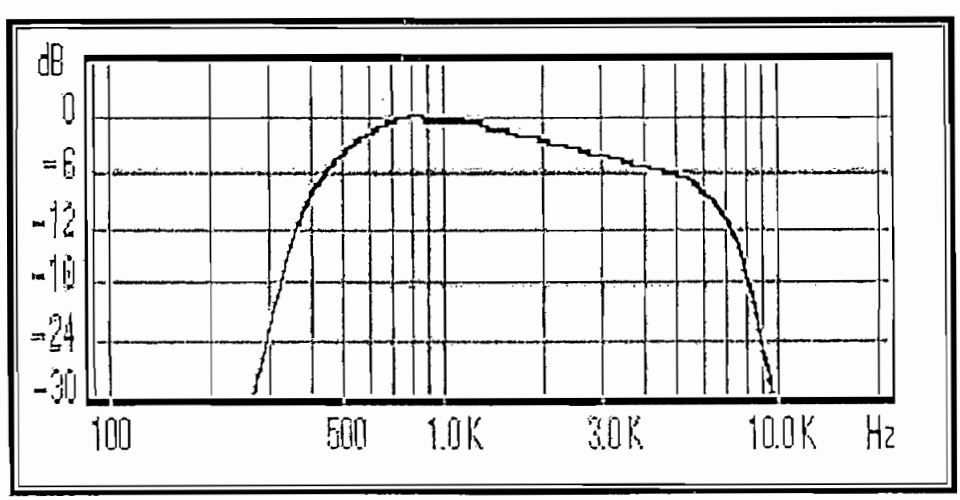


FIGURA 1.39) Respuesta de frecuencia de un altavoz

En la figura anterior se puede notar que frecuencias menores a 450 Hz y mayores a 10 KHz no serán reproducidas por este altavoz debido a que no tiene respuesta para estas frecuencias. Sin embargo aunque éste altavoz tiene respuesta en el rango de 450 a 10000 Hz esta no es uniforme para todo este rango, es decir es variable la respuesta (ganancia) para diferentes frecuencias.

Los extremos del rango de frecuencias en el cual el dispositivo de audio tiene respuesta son fijados a –3dB del nivel que se podría asumir como constante en este rango.

1.8.3 SLEW RATE

Es una medida de la rapidez con la que una etapa de potencia puede responder a un cambio brusco de la tensión en la entrada. Las unidades de esta medida son voltios por unidad de tiempo (V/s), aunque se suele expresar en V/µs (voltios / microsegundo). Esta medida nos dice exactamente cuantos voltios puede aumentar la tensión de salida en un microsegundo sin que haya deformación.

Cuanto mayor sea el valor del Slew rate del equipo, mejor será éste. El problema que se genera cuando el equipo tiene un slew rate insuficiente, es que no puede seguir las variaciones grandes de señal, provocando el efecto de triangulación, es decir, deformando la señal y generando distorsión.

Este efecto de triangulación, se producirá cuando el equipo trabaje a alta potencia, ya que es ahí donde se le exigen grandes variaciones de la tensión de salida.

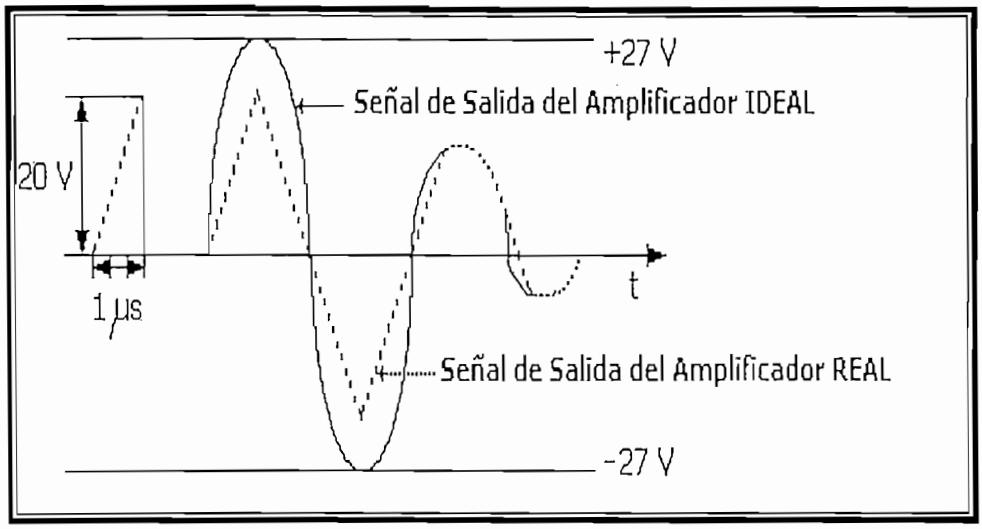


FIGURA 1.40) Efecto de triangulación por un Slew rate pequeño

En la figura 1.40 se muestra una señal con línea continua que representa una forma de onda hipotética que tiene que presentar el amplificador de potencia a la salida, y en línea punteada la forma de onda que muestra al estar limitado el valor del slew rate y no poder seguir esa onda.

El problema se hace más evidente en las grandes excursiones de tensión (al principio de la forma de onda). El amplificador eleva la tensión de salida lo más rápido que puede (20 V/µs), pero no es suficiente para seguir la forma de onda, en el caso extremo, describe una forma de onda triangular.

1.8.4 SENSIBILIDAD DE ENTRADA

Es el mínimo valor de la tensión en voltios que hay que aplicar a la entrada de la etapa de potencia, para que este pueda reproducir y entregar a la carga la potencia nominal.

La sensibilidad es una medida de calidad, cuanto más sensible sea la etapa de potencia, mayor calidad tendrá. La sensibilidad de entrada en una etapa de potencia, equivale a la agudeza auditiva de un oyente; si tiene poca sensibilidad, es duro de oídos.

La sensibilidad se mide de la siguiente forma: con el control de nivel a máxima potencia, se va aumentando el nivel de tensión de la señal sinusoidal de entrada de 1000 Hz, hasta que a la carga (parlante) se le entregue la potencia nominal.

La potencia consumida en la carga se puede calcular porque se conoce el valor de la misma (R Ω) y se mide la tensión en bornes (V_{rms}), así la potencia nominal en vatios es: $P = V_{rms}^2/R$.

1.8.5 IMPEDANCIA DE ENTRADA

Es la resistencia más la reactancia que "ve" el equipo anterior. Para análisis, la impedancia de entrada se la toma solo como un valor resistivo, y los valores que normalmente se encuentran para amplificadores de potencia están entre 10 y 50 KΩ. En audio, lo que se busca es cumplir el principio de máxima transferencia de energía, es decir, impedancias de salida de los aparatos muy bajas (en el orden de centenas de ohmios) e impedancias de entrada del amplificador muy altas (en el orden de decenas de miles de ohmios).

1.8.6 IMPEDANCIA DE SALIDA

Es la resistencia más la reactancia que "ve" el equipo posterior a la etapa. De igual manera que para la impedancia de entrada para análisis se la toma solo como un valor resistivo. Esta característica hace que parte de la potencia generada se consuma en la salida misma del amplificador.

El criterio de adaptación en tensión busca que la impedancia de salida del amplificador sea la menor posible y la de la carga (parlante) sea la mayor posible. De este modo la mayor parte de la potencia se consumirá en el altavoz. La

intensidad de corriente es la misma para las dos cargas, puesto que están en serie, por lo tanto, la de mayor valor consumirá más potencia.

1.8.7 FACTOR DE AMORTIGUAMIENTO (FA)

El factor de amortiguamiento o conocido también como damping factor (DF), es la relación entre la impedancia de la carga y la impedancia de salida del amplificador:

$$FA_{teorico} = \frac{Resistencia\ de\ Carga}{Resistencia\ de\ Salida} \qquad \text{Ec. 1.71}$$

Por impedancia de la carga se entiende exclusivamente la impedancia del parlante. Se suele tomar el valor nominal, que es sólo resistivo. Usualmente, lo que hacen muchos fabricantes, es dar el Damping Factor para un valor concreto de Resistencia de carga. Por ejemplo, FA=150 para $R_{carga} = 8\ \Omega$ a 1kHz. Con lo cual se puede despejar que la impedancia de salida a 1kHz es $8/150 = 0.053\ \Omega$.

Existen distintas formas de presentar el mismo dato, la impedancia de salida. $FA = DA = 150$ significa que la carga consume 150 veces lo que consume la salida de la etapa o que la salida consumirá aproximadamente 1/150 de la potencia total.

La medida teórica que realiza el fabricante no incluye la resistencia que añade el cable, lo que es normal, puesto que eso dependerá de la instalación final. Cuando existe cable entre la carga (parlante) y la salida de la etapa, la resistencia del cable debe añadirse a la impedancia de salida de la etapa para obtener un valor del Factor de Amortiguamiento mas real.

$$FA_{real} = \frac{Resistencia\ de\ Carga}{Resistencia\ de\ Salida + Resistencia\ del\ Cable} \qquad \text{Ec. 1.72}$$

Un cable malo, tendrá un valor de resistencia alto, que se multiplicará por los metros de cable, haciendo que disminuya el Factor de Amortiguamiento, es decir, a la carga le llegará menos potencia. Si el cable tiene una resistencia total de 1Ω, se tendría que FA = 8/(0.053+1) que es casi lo mismo que 8/1, con lo que FA = 8; lo que significa que de cada nueve partes de potencia, una se consume antes de llegar a la carga y ocho en la carga. De 100W, sólo 88.8 se consumen en la carga.

Si se hace un análisis del FA para altas frecuencias, las impedancias tanto de carga como de salida serán compuestas de una parte resistiva y una parte de reactancia, siendo en mayor grado para el caso de la impedancia de carga, ya que las cargas que usualmente son parlantes están compuestos por bobinas que sumarian una reactancia inductiva a la impedancia de carga, de tal manera que se produciría un aumento en el valor de FA. En conclusión este factor afecta a todas las frecuencias, pero en mayor grado a las bajas frecuencias.

1.8.8 RENDIMIENTO

Este dato proporciona la cantidad de energía que entrega a su salida (cable + parlantes) la etapa de potencia, con respecto a la que en realidad se consume. El rendimiento se calcula:

$$\text{Rendimiento } \% = \frac{P_{entregada}}{P_{consumida}} * 100 \qquad \text{Ec.1.73}$$

La parte de energía que no sale se consume en forma de calor. La mayor parte de este calor se produce en la etapa de potencia, que es la que maneja grandes tensiones e intensidades. Para que el exceso de calor no produzca averías en el equipo, se emplean disipadores de calor y ventiladores para forzar el flujo de aire.

1.9 PREAMPLIFICADORES Y MEZCLADORES

1.9.1 PREAMPLIFICADORES

Los circuitos transductores de energía, generadores de señales eléctricas como son el micrófono, casete, CD, etc., tienen la desventaja de entregar a la salida señales que trabajan en el orden de los milivoltios.

Los niveles de salida de los diferentes equipos se muestran en la siguiente tabla:

EQUIPO	NIVEL DE SALIDA
Micrófono (600 Ω)	1 mV
Micrófono (10 KΩ)	2.5 mV
Casetera, CD, DVD, DAT, etc.	150 mV

TABLA 1.4) Niveles de voltaje entregados por distintas fuentes

Por otro lado los circuitos amplificadores de audio, no pueden trabajar con señales de niveles muy bajos a la entrada, puesto que la amplificación depende de las resistencias de retroalimentación las mismas que no deben sobrepasar los KΩ , debido a que resistencias de valores mayores, son muy ruidosas y en los sistemas de audio es lo menos que se quiere obtener.

Presentándose este par de situaciones, se necesitan circuitos intermedios que sirvan de interfaz entre las señales de bajo nivel provenientes de los equipos antes mencionados y las señales adecuadas que deben ingresar a los amplificadores de audio.

Estos circuitos de interfaz, cuyo trabajo es aumentar el nivel de la señal, toman el nombre de Preamplificadores, que como su nombre lo explica es una

etapa previa a la amplificación. Estos circuitos de preamplificación, pueden constar de una o varias etapas, dependiendo del requerimiento.

La estructura de estos preamplificadores es igual a la de cualquier amplificador de voltaje, como son los amplificadores Inversores y No Inversores que se desarrollan basándose en amplificadores operacionales.

1.9.1.1 Amplificadores de Voltaje

Son circuitos capaces de entregar a la salida, señales de voltaje que son semejantes en forma a las señales de entrada, pero con niveles de amplitud superiores.

Estos amplificadores de voltaje pueden ser construidos en base a amplificadores operacionales y según la señal que entregan a la salida estos pueden clasificarse en dos, que se los muestra a continuación :

1.9.1.1.1 Amplificador No Inversor

Estos son amplificadores de voltaje (figura 1.41) con una señal de salida idéntica en forma a la de la entrada cuya ganancia de voltaje en lazo cerrado está dada por la expresión:

$$\frac{V_{out}}{V_{In}} = 1 + \frac{R_2}{R_1} = A_{vf} \qquad \text{Ec. 1.74}$$

La ganancia toma en cuenta la operación del circuito Retroalimentado, la misma que es una ganancia estabilizada, como ya se mencionó anteriormente en el literal 1.2.1.4.

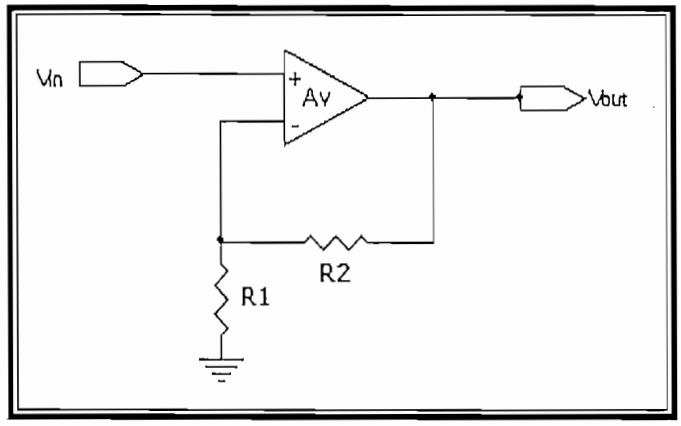


FIGURA 1.41) Amplificador de voltaje No inversor

La impedancia de entrada como la de salida para un circuito amplificador de voltaje No Inversor (figura 1.42), con retroalimentación RSDP está dada por:

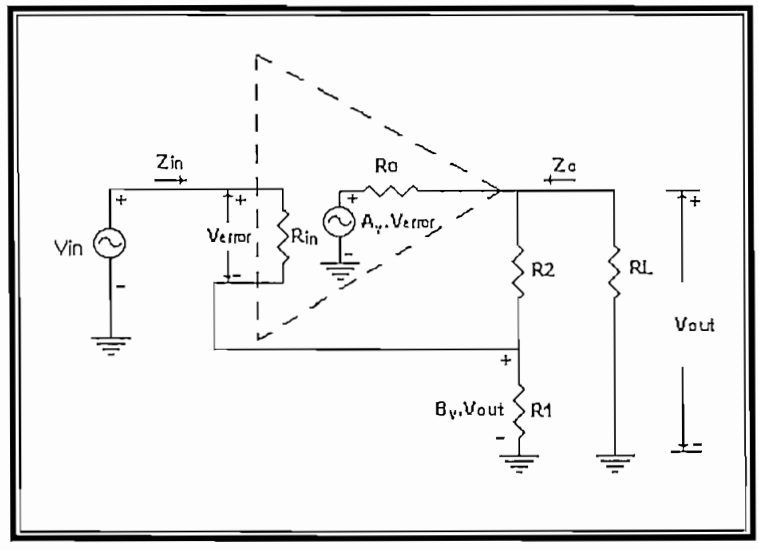


FIGURA 1.42) Diagrama que muestra las impedancias en los diferentes puntos de un amplificador No inversor.

El amplificador operacional tiene una impedancia de entrada en lazo abierto aproximadamente de R_{In}. Todo el amplificador tiene una impedancia de entrada en lazo cerrado de Z_{In}. La impedancia en lazo cerrado Z_{in} es mayor que la impedancia en lazo abierto R_{In}.

Para encontrar la impedancia de entrada en lazo cerrado *se parte* de la siguiente expresión:

$$Z_{In} = \frac{V_{In}}{I_{In}} \qquad \text{Ec. 1.75}$$

En donde:

$$V_{in} = V_{error} + B_v \cdot V_{out}$$

$$V_{in} = V_{error} + A_v \cdot B_v \cdot V_{error}$$

$$V_{In} = (1 + A_v \cdot B_v) \cdot V_{error} \qquad \text{Ec. 1.76}$$

Como $V_{error} = I_{In} \cdot R_{In}$ se reemplaza en la ecuación 1.76 se tiene lo siguiente:

$$Z_{In} = (1 + B_v \ A_v \ ) \cdot R_{In} \qquad \text{Ec. 1.77}$$

Asi mismo se puede visualizar en la figura 1.42 que este tipo de amplificadores de voltaje tiene una impedancia de salida en lazo abierto R_o y una impedancia en lazo cerrado Z_o.

La impedancia de salida en lazo cerrado es inferior a la impedancia de salida en lazo abierto y su valor se lo encuentra de acuerdo a la ecuación 1.22 con lo que se tiene:

$$Z_o = \frac{R_o}{1 + A_v . B_v} \qquad \text{Ec 1.78}$$

1.9.1.1.2 Amplificador Inversor

Son amplificadores de voltaje con la particularidad de entregar como salida una señal amplificada invertida en comparación con la de entrada. La ganancia de voltaje en lazo cerrado de este tipo de amplificador, esta dada por:

$$\frac{V_{out}}{V_{In}} = -\frac{R_3}{R_1} = A_{vf} \qquad \text{Ec 1.79}$$

El diagrama de un amplificador Inversor se representa en la figura 1.43:

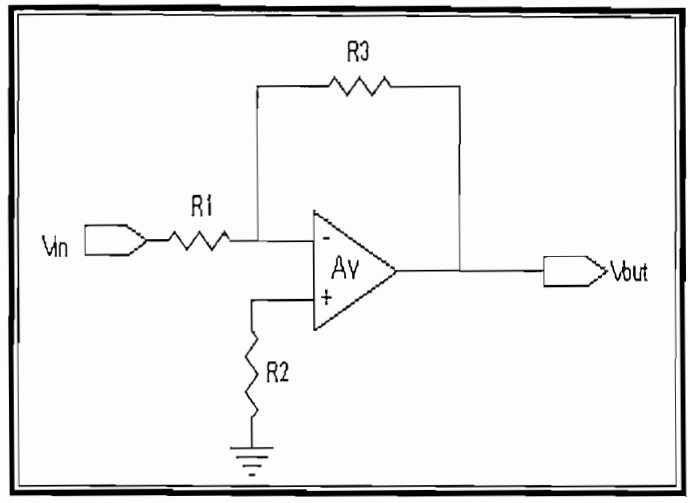


FIGURA 1.43) Circuito amplificador de voltaje Inversor

Este tipo de amplificador de voltaje tiene una realimentación negativa del tipo RPDP (Retorno Paralelo Derivación Paralelo).

Para encontrar el valor de la impedancia de entrada como la de salida nos basamos en el teorema de Miller, en el cual la idea es dividir el condensador de realimentación de un amplificador en dos, uno a la entrada y otro a la salida. De la misma manera se utiliza este teorema es este caso, con la diferencia que aqui se tiene una resistencia de Miller de realimentación la cual se la puede dividir en una resistencia Miller a la entrada y otra a la salida, tal como se muestra en la figura 1.44.

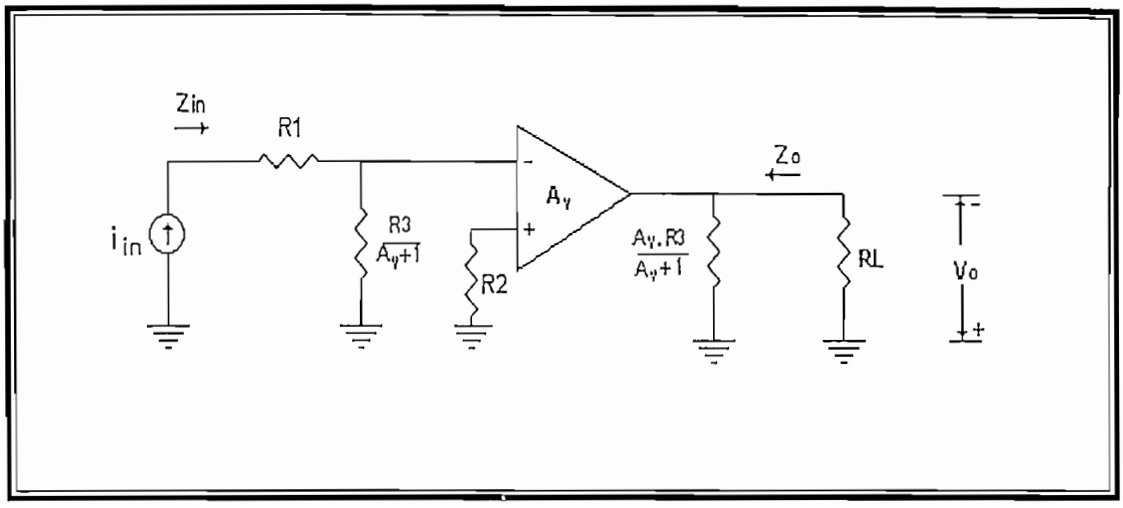


FIGURA 1.44) Circuito amplificador de voltaje Inversor

Según la figura 1.44 se establece que la impedancia de entrada para un amplificador inversor está dada por la siguiente expresión:

$$Z_{In} = R_1 + \frac{R_3}{A_v + 1} \qquad \text{Ec 1.80}$$

Para el caso de la impedancia de salida se tiene lo siguiente:

$$Z_o = \frac{A_v \cdot R_3}{A_v + 1} \qquad \text{Ec 1.81}$$

Si la ganacia en lazo abierto del amplificador operacional es muy grande se tiene lo siguiente:

$$Z_{In} = R_1 \qquad \text{Ec 1.82}$$

$$Z_o = R_3 \qquad \text{Ec 1.83}$$

1.9.2 MEZCLADORES

Generalmente en los sistemas de audio no se tiene solamente una sola señal de entrada, por el contrario se tienen dos o más señales de entrada, sin embargo la amplificación de potencia se realiza de una sola señal de entrada por lo que es necesario mezclar o sumar todas las señales de entrada existentes.

Los circuitos encargados de realizar esta operación se llaman circuitos sumadores o mezcladores, que además también amplifican esta señal para luego enviarla a la etapa de amplificación de potencia. Estos circuitos pueden ser Inversores y No Inversores, a continuación se representan estos circuitos en forma ideal.

1.9.2.1 Mezclador No Inversor

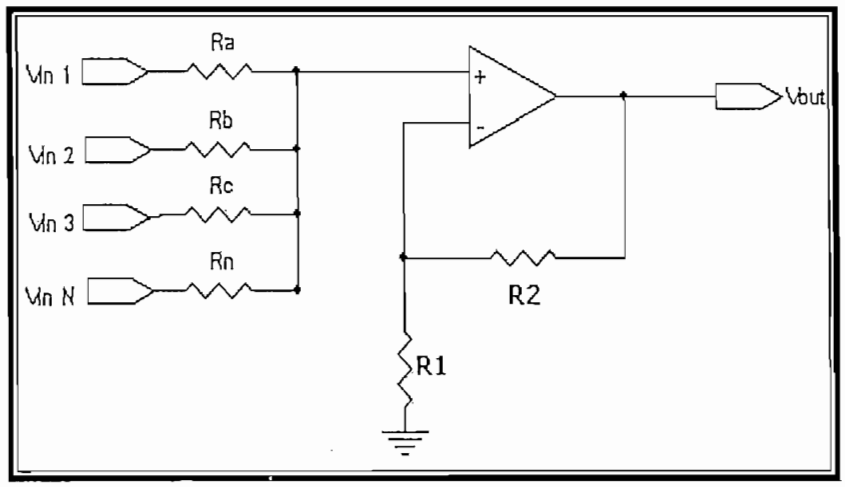


FIGURA 1.45) Circuito mezclador No Inversor

$$V_{OUT} = R_a \,||\, R_b \,||\, R_c \,||\, R_n \left(\frac{V_{in1}}{R_a} + \frac{V_{in2}}{R_b} + \frac{V_{in3}}{R_c} + \frac{V_{inN}}{R_n} \right) \left(1 + \frac{R_2}{R_1} \right) \qquad \text{Ec. 1.84}$$

1.9.2.2 Mezclador Inversor

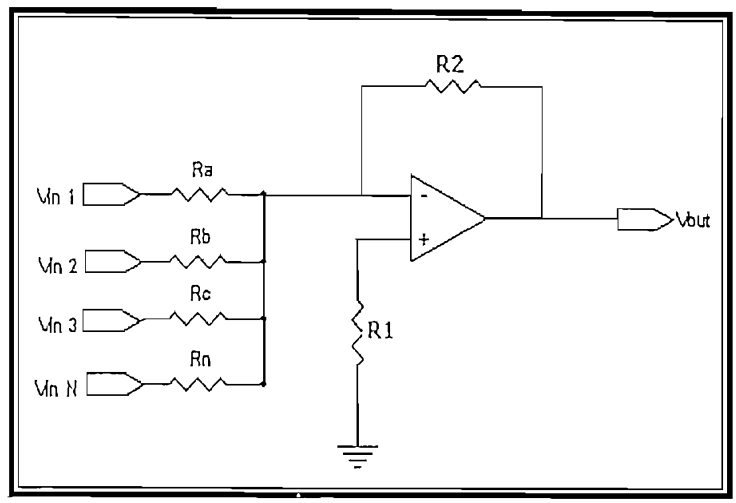


FIGURA 1.46) Circuito mezclador inversor

$$V_{OUTY} = -R_2\left(\frac{Vin_1}{R_a} + \frac{Vin_2}{R_b} + \frac{Vin_3}{R_c} + \frac{Vin_N}{R_n}\right) \qquad \text{Ec. 1.85}$$

1.10 PROTECCIONES DE DISPOSITIVOS Y DE CIRCUITOS

El trabajar con dispositivos electrónicos que se manejan con ciertas limitaciones propias de su estructura, tanto para funcionar correctamente, como para no destruirse, nos obliga a precautelar estas situaciones y utilizar diversos mecanismos a los cuales se les llama "protecciones".

El trabajar con dispositivos que no son totalmente perfectos, que generan pérdidas que se transforman en calor, el cual debe ser disipado en forma suficiente y eficaz, a fin de que el dispositivo opere por debajo de su límite superior de temperatura.

De la misma manera existen algunas circunstancias que se deben tomar en cuenta para garantizar el funcionamiento individual y colectivo de un circuito electrónico.

A continuación se presentan algunos mecanismos de protección, tanto para dispositivos como para circuitos que serán utilizados en el presente proyecto.

1.10.1 ENFRIADORES Y DISIPADORES DE CALOR

Debido a las pérdidas por operación y por conmutación, dentro del dispositivo, especialmente de los que trabajan con altos valores de potencia, se genera calor. Este calor debe ser transferido del dispositivo a un medio más frío, a fin de mantener la temperatura de operación del dispositivo dentro del rango especificado.

Esta transferencia de calor puede llevarse a cabo mediante conducción, convención o radiación, ya sea natural o de aire forzado. Este calor debe fluir desde el dispositivo a su carcasa y de ahí hasta el disipador de calor en el medio enfriador.

A continuación se presenta un circuito eléctrico que representaría un sistema de disipación.

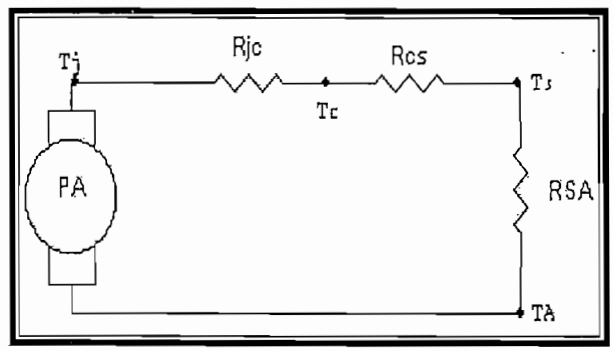


FIGURA 1.47) Circuito eléctrico equivalente de un sistema de disipación.

Según la figura 1.47, P_A es la perdida de potencia promedio del dispositivo y con la temperatura de la unión de un dispositivo T_j dada por:

$$T_j = P_A\left(R_{JC} + R_{CS} + R_{SA}\right) + T_A \qquad \text{Ec. 1.86}$$

En donde: R_{JC} = resistencia térmica de la unión a la carcasa, °K/W

R_{CS} = resistencia térmica de la carcasa al disipador, °K/W

R_{SA} = resistencia térmica del disipador al ambiente, °K/W

T_A = temperatura ambiente

R_{JC} y R_{CS} son valores que por lo general son especificados por el fabricante del dispositivo. Una vez que se conoce la pérdida de potencia del dispositivo, P_A, se puede calcular la resistencia térmica requerida del disipador de calor asumiendo un valor de Tj según las hojas de datos del fabricante del elemento y de acuerdo a las condiciones con las que se desea trabajar. El siguiente paso es seleccionar el disipador de calor de una dimensión que cumpla con el requisito de la resistencia térmica.

Hay una amplia variedad de disipadores de calor de aluminio extruido disponible comercialmente, que utilizan aletas de enfriamiento a fin de aumentar la transferencia de calor.

1.10.2 TRANSITORIOS DEL LADO DE LA ALIMENTACIÓN Y DEL LADO DE LA CARGA

Usualmente en circuitos convertidores de señal alterna a continua se utiliza a la entrada un transformador, para los cuales bajo una operación normal, se almacena una cantidad de energía en la inductancia de magnetización L_m y al retirar la alimentación se produce un voltaje transitorio a la entrada.

A fin de eliminar el voltaje transitorio, se puede conectar un capacitor a través del primario o del secundario del transformador, tal como se muestra en la figura 1.48; en la práctica también se conecta una resistencia en serie con el capacitor, con el objetivo de limitar la oscilación transitoria de voltaje. Un gráfico del circuito equivalente durante la desconexión se muestra en la figura 1.49.

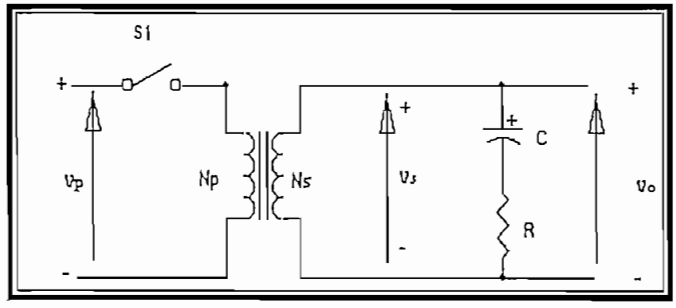


FIGURA 1.48) Capacitor y resistencia conectados en el secundario para eliminar voltaje transitorio

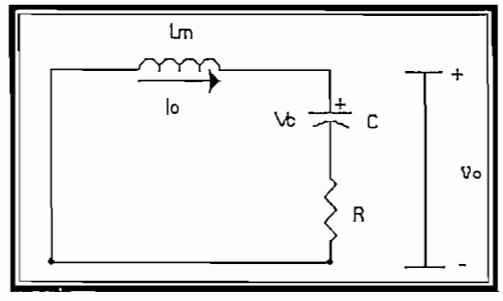


FIGURA 1.49) Circuito equivalente en el momento de la desconexión

En el momento de la desconexión se tiene:

$$Vc = Vm.Sen\ \theta \qquad \text{Ec.1.87}$$

$$Io = \frac{Vm}{\omega\, Lm} \cos\ \theta \qquad \text{Ec. 1.88}$$

En donde: V_c = voltaje en el capacitor al momento de la desconexión

$V_m = \sqrt{2}V_s$ valor máximo del voltaje de entrada

I_o = corriente de magnetización

$\omega = 2nf$

L_m = inductancia de magnetización

Con lo que desarrollando un análisis matemático se llega a que:

$$C = \frac{Io}{Vp\ .\omega\ o} \qquad \text{Ec. 1.89}$$

Donde V_P es el voltaje transitorio máximo que se quiere eliminar.

De la misma manera ocurre en el lado de la carga, para lo cual también se utiliza el circuito y los conceptos antes vistos, así como se muestra en la figura 1.50:

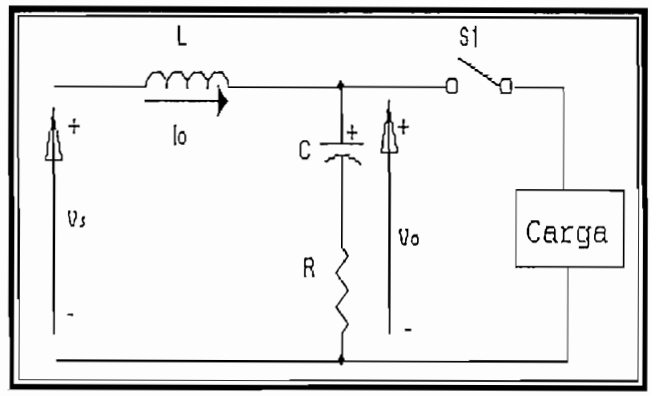


FIGURA 1.50) Capacitor y resistencia conectados en paralelo a la carga para eliminar voltaje transitorio

1.10.3 PROTECCIONES CONTRA SOBRE CORRIENTE

Los convertidores de potencia pueden provocar cortos circuitos o fallas, y las corrientes resultantes deberán eliminarse con rapidez. Normalmente se utilizan fusibles de acción rápida a fin de proteger a los dispositivos semiconductores. Conforme aumente la corriente de falla, el fusible se abre y elimina el problema en unos cuantos milisegundos.

La protección de estos dispositivos semiconductores, depende también de la selección de la localización de los fusibles, sin embargo, los fabricantes de fusibles recomiendan que se coloque el fusible en serie con cada uno de los dispositivos.

En la selección del fusible adecuado para la protección de un cierto dispositivo, se debe tomar en cuenta los valores límites de corriente que puede soportar para no destruirse. Además como punto importante en la selección, se debe tomar en cuenta la velocidad de respuesta del fusible a la sobrecorriente, así como también las curvas corriente-tiempo del dispositivo.

1.10.4 PROTECCIONES CONTRA TENSIÓN CONTINUA

Este tipo de protecciones se las debería utilizar en todos los circuitos electrónicos que trabajan con elementos activos (es decir que necesitan un voltaje de polarización para funcionar) para proteger su integridad.

Se las utiliza a la entrada de los circuitos para evitar que cualquier voltaje DC proveniente de la fuente de alimentación trate de circular hacia la entrada, esto producto de algún error o desperfecto. Si ocurriere algunos de estos errores o desperfectos y no se dispone de este tipo de protección se ocasionaría un cortocircuito a la entrada averiado al generador de la señal.

Este tipo de protecciones se basa en los principios de los condensadores, los cuales trabajan como circuito abierto cuando se les aplica señales continuas. Por el contrario, si se está trabajando con una señal con una frecuencia diferente de cero, que es una señal alterna, el condensador después de haberse cargado funciona como corto circuito.

Con este mismo concepto se trabaja en circuitos acoplados en donde se debe evitar que existan corrientes inversas de un circuito a otro.

1.11 DISEÑO DEL SISTEMA DE AUDIO

El problema planteado es construir un sistema de audio de dos canales de salida y cuatro entradas. Cada canal debe ser capaz de entregar 25 W de potencia nominal por canal como máximo, las cuatro entradas consisten en una de tipo monofónico para micrófono (*IN MIC2*) y tres de tipo estereo, una para micrófono (*IN MIC1*) y dos de tipo estereo para entradas auxiliares (*IN AUX1 e IN AUX2*).

Las entradas de micrófono (*IN MIC)* deben soportar señales de micrófonos de cualquier impedancia. Las entradas auxiliares (*IN AUX)* son para el ingreso de señales provenientes de caseteras, CD's, DVD's, etc.

El diagrama de bloque del sistema de audio correspondiente a un canal constará de los siguientes bloques:

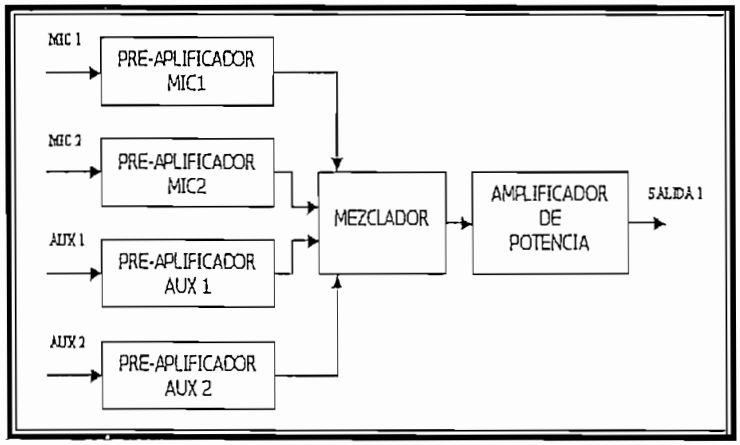


FIGURA 1.51) Diagrama de bloques de un sistema de audio

1.11.1 DISEÑO DE LOS PREAMPLIFICADORES

1.11.1.1 Requerimientos y especificaciones

Los preamplificadores como ya se mencionó, sirven para elevar los bajos niveles de voltaje que provienen de las entradas (micrófonos, caseteras, CD's, etc.), ya que los amplificadores de potencia tienen una cierta sensibilidad y si la señal de entrada no esta dentro de este margen de sensibilidad no se realizará la amplificación del sonido.

Por el lado de la sensibilidad de la etapa de preamplificación no hay inconvenientes ya que los CI´s que se utilizan para este trabajo tienen valores de sensibilidad que tienden a cero en sus entradas (unidades de milivoltios).

Se debe tomar en cuenta también que esta etapa sirve de acoplamiento entre las entradas y el amplificador, por lo que debe tener una alta impedancia de entrada y una baja impedancia de salida de tal manera que se produzca la máxima transferencia de energía entre las entradas y la etapa de amplificación.

Esta etapa es muy importante y se debe tratar de tener la menor distorsión posible.

1.11.1.2 Diseño de Preamplificadores para Micrófonos

Para fijar el valor de la señal de salida de estos preamplificadores, tomamos en cuenta el valor de la señal que ingresa ya sea para los micrófonos como para las entradas auxiliares.

El valor de la señal proveniente de una entrada auxiliar está en el orden de los 150 mV, que esta dentro del margen de sensibilidad que usualmente tienen los amplificadores de audio, (por ejemplo: STK4141) por lo que no sería necesario utilizar una etapa de preamplificación para aumentar el nivel de voltaje. Sin embargo esta etapa es utilizada para realizar el acoplamiento de impedancias.

Por el lado de las señales provenientes de los micrófonos, se tiene que está en el orden de 1 mV, la mínima, valor que es 150 veces menor que la señal proveniente de una entrada auxiliar.

Como nivel de salida de los preamplificadores se escoge un valor mayor al que se tienen en las entradas auxiliares, para garantizar que la señal de salida de los preamplificadores sea mayor al valor de sensibilidad de la etapa de potencia. De acuerdo a esto se escoge el valor de 300 mV para las salidas de los preamplificadores que esta en el rango de los requerimientos antes citados.

Este valor es mayor que 150 mV, que fácilmente se lo puede obtener con un amplificador inversor. Para las entradas de micrófonos se necesitará una etapa de amplificación de 300, que usualmente se suele obtener mediante dos etapas, pero por las características del CI con el que se va a trabajar (TL084) se lo puede realizar con una sola etapa solamente sin tener deformaciones en la señal amplificada.

Para tener los niveles de voltajes requeridos a la entrada del amplificador y de esta forma entregar una potencia de 25 Vatios, se coloca a la salida de cada preamplificador un potenciómetro el cual permita tomar solo una parte de la señal de salida.

Para las señales provenientes de los micrófonos se tienen dos valores 1 mV y 2.5 mV como ya se vio anteriormente. Entre estos valores no existe mayor margen de diferencia, por lo que se va a realizar un solo diseño, tomando como referencia el menor valor correspondiente a 1 mV.

A continuación presentamos un esquemático del boceto del circuito a diseñarse:

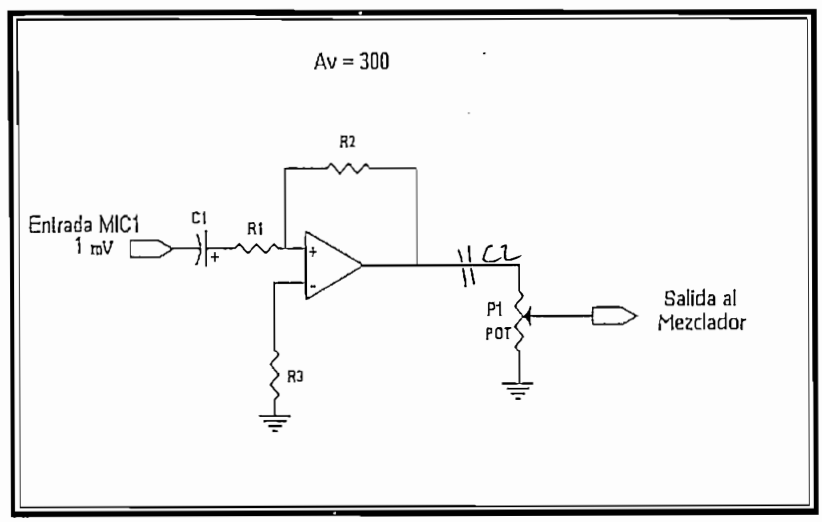


FIGURA 1.52) Circuito preamplificador para una entrada de micrófono

Este preamplificador consta de una sola etapa, como lo muestra la figura anterior; obteniéndose una ganancia total de 300, lográndose amplificar la señal de 1 mV proveniente de la entrada de micrófono a una señal de 300 mV que es la que va a ingresar al mezclador.

El capacitor C_1 se lo utiliza para aislar la componente continua de la entrada. De igual manera el capacitor C_2 sirve bloquear la componente continua a la salida.

- Diseño

Datos:

$V_{In} = 1\,mV$

$V_{out} = 300\,mV$

$A_v = 300$

Se asume $R_1 = 1K\Omega$ y aplicando la expresión Ec. 1.79 correspondiente a un amplificador Inversor se tiene que:

$$R_2 = 300\,K\Omega\ (Ideal)$$

$$R_2 = 300\,K\Omega$$

El valor de la resistencia R_3 es igual al paralelo de las resistencias R_2 y R_1 es decir:

$$R_3 = 996.67\,\Omega\ (Ideal)$$

$$R_3 = 910\,\Omega$$

Para calcular el valor del capacitor C_1, se debe tomar en cuenta que $Z_{In1} >> X_{C1}$ para que no haya pérdida de señal en X_{C1}.

Por las condiciones propuestas se puede asumir que $X_{C1} = 10^3$, cuyo valor cumple la condición antes mencionada y si aplicamos la siguiente expresión podemos encontrar el valor de C_1.

$$X_{C1} = \frac{1}{2 \cdot \pi \cdot f \cdot C_1} \qquad \text{Ec. 1.90}$$

En donde f es igual a 20 Hz.

Despejando de la ecuación 1.90 se obtiene la expresión para calcular el valor de C_1:

$$C_1 = \frac{1}{2 \cdot \pi \cdot f \cdot 1 \times 10^{3}} \qquad \text{Ec.1.91}$$

$$C_1 = 7.95\ \mu F$$

$$C_1 = 10\ \mu F$$

El diseño anterior efectuado es exactamente igual para la otra entrada de micrófono.

1.11.1.3 Diseño de Preamplificadores para Entradas Auxiliares

Estos preamplificadores constan de una sola etapa y el esquemático que muestra el circuito es el siguiente:

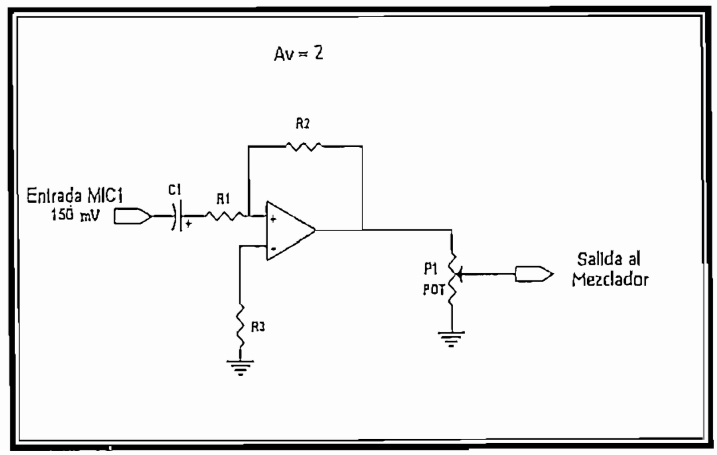


FIGURA 1.53) Circuito de preamplificador para una entrada auxiliar

Datos:

$V_{In} = 150\ mV$

$V_{out} = 300\ mV$

$A_v = 2$

Asumiendo el valor de $R_1 = 1K\Omega$ y aplicando la expresión Ec.1.79 se tiene que:

$$R_2 = 2\ K\Omega\ (Ideal)$$

$$R_2 = 2.2\,K\Omega$$

El valor de R_3 sería:

$$R_3 = 687.5\,\Omega\ (Ideal)$$

$$R_3 = 680\,\Omega$$

Los valores de los capacitores C_1 y C_2 son iguales a los encontrados en el diseño anterior.

$$C_1 = 10\,\mu F$$

$$C_2 = 0.1\,\mu F$$

Este mismo diseño se lo utiliza para la otra entrada auxiliar.

1.11.2 DISEÑO DEL AMPLIFICADOR DE POTENCIA

1.11.2.1 Requerimientos y especificaciones

El objetivo es diseñar un amplificador de potencia para audio (20Hz a 20KHz) de dos canales capaz de entregar 25 vatios de potencia máxima por canal indistintamente de la carga que se conecte, ya sea ésta de 4, 6 u 8 ohmios.

Este amplificador debe estar conformado de las etapas antes mencionadas: etapa de entada, etapa excitadora (driver), etapa de potencia y realimentación. Como una etapa adicional se debe incluir un circuito contra cortocircuito o sobrecarga en la carga. El diseño se lo debe realizar para la carga con la cual se requiera mayor consumo de corriente.

El control del amplificador para que sea capaz de efectuar este trabajo, se lo realiza mediante computador a través de una Tarjeta de Adquisición de Datos y varios potenciómetros digitales con la circuitería correspondiente, cuyos detalles se los mencionará posteriormente.

Como se menciono anteriormente la potencia que entrega el amplificador la obtiene de la fuente de alimentación, por lo que esta debe de estar bien dimensionada tanto en el voltaje, como en la corriente que va a entregar para poder obtener 25 W o más.

1.11.2.2 Diseño del amplificador:

- **Datos**

  Potencia máxima de salida por canal (P_o) = 25 W

  Resistencias de Carga (R_L) = 4 ohmios.

El circuito que representa el diseño por canal a seguirse se muestra en la figura 1.54.

- **Etapa de potencia:**

Se ha seleccionado los siguientes Transistores para que trabajen en esta etapa y sus características son las siguientes:

Elemento	Tipo	V_{CB_o} (V)	V_{CE_o} (V)	V_{EB_o} (V)	I_{Cmax} (A)	P_{Dmax} (W)	F_t (MHz)	β
ECG 130	NPN	180	180	5	16	150	6	70
ECG 291	NPN	130	120	5	4	40	4	75
ECG 292	PNP	130	120	5	4	40	4	75

TABLA 1.5) Características de transistores

Se debe tomar en cuenta las siguientes expresiones:

$$Po_{max} = V_{OUT\ max} \cdot I_{L\ max} \quad \text{y} \quad I_{L\ max} = \frac{V_{OUT\ max}}{RL} \qquad \text{Ec. 1.92 y Ec. 1.93}$$

Con lo que se obtiene que la potencia de salida está dada por:

$$Po_{max} = \frac{V_{OUT\ max}^{\ 2}}{RL} \quad \text{y} \quad Po_{max} = I_{L\ max}^2 \cdot RL \qquad \text{Ec. 1.94 y Ec. 1.95}$$

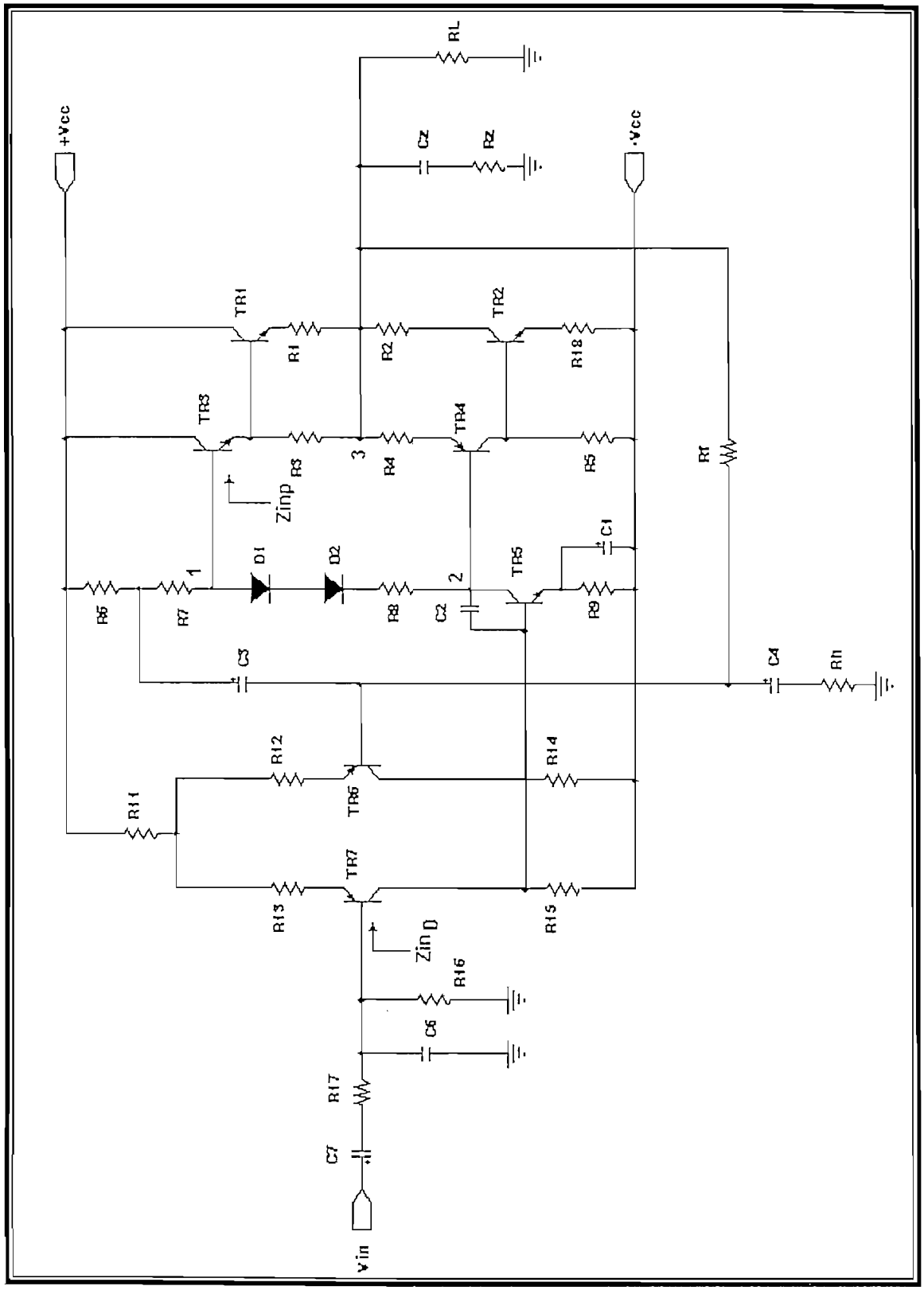


FIGURA 1.54) Circuito Amplificador de Audio a diseñarse

Despejándose la expresión anterior se tendrá que el voltaje y la corriente de salida para cada par de transistores (Darlington y Retroalimentado) es igual a:

$$V_{OUT\ max} = \sqrt{2.\ Po_{max}\ .\ RL} \qquad \text{Ec. 1.96}$$

$$I_{L\ max} = \sqrt{\frac{2.\ Po_{\ max}}{R_L}} \qquad \text{Ec. 1.97}$$

Para los datos dados se obtiene los siguientes valores:

$$V_{OUTmax} = 14.14\ V$$
$$I_{Lmax} = 3.53\ A$$

Para limitar la corriente que va hacia la carga se coloca las resistencias R_1 y R_2 en el emisor y colector de T_{R1} y T_{R2} respectivamente, estas resistencias limitadoras además proporcionan estabilidad térmica. Se asume que el voltaje que cae en estas resistencias es muy pequeño ($V_{R1} = V_{R2} = 1\ V$), para que no haya perdida de señal a la salida.

$$R_1 = R_2 = \frac{V_{R1}}{I_{L\,max}} = \frac{1}{3.53} = 0.28\Omega$$

Se escoge el valor comercial de $0.33\ \Omega$.

Tomándose en cuenta que por R_1 y R_2 circula una señal únicamente durante medio ciclo, la potencia que deben soportar estas resistencias es la siguiente:

$$V_{R1} = R_1 \cdot I_{Lmax} = 0.33\Omega \times 3.53A = 1.16V$$

$$P_{R1} = P_{R2} = \frac{(V_{R1}/\ 2)^2}{R_1} = 1.03\ W$$

Entonces:

$$R_1 = 0.33\,\Omega / 5\,W$$

$$R_2 = 0.33\,\Omega / 5\,W$$

Se utilizan los transistores ECG130 en T_{R1} y T_{R2} por tener una gran ganancia de corriente. La corriente de Base en T_{R1} y T_{R2} es la siguiente:

$$I_{BTR1} = I_{BTR2} = \frac{I_{ETR1}}{\beta_{TR1} + 1} = \frac{3.53\,A}{71} = 49.71\,mA$$

Para que los transistores T_{R3} y T_{R4} trabajen en clase AB se escoge la siguiente condición:

$$I_{CTR3} = \frac{I_{BTR1}}{100}$$

Entonces:

$$I_{CTR3} \approx I_{ETR3} = \frac{49.71\ mA}{100} = 0.497\ mA$$

Para limitar la corriente que ingresa a T_{R1} y a T_{R2} se coloca las resistencias R_3 y R_5, se asume que la corriente que pasa por estas resistencias es mayor que la corriente que ingresa a las bases de T_{R1} y T_{R2}. Para hacer cumplir este requerimiento se debe seguir la siguiente condición:

$$R_3 < Zin_{TR1}$$

Con lo que se tendrá:

$$R_3 < \left|(\beta_{TR1} + 1).(r_{e1} + R_1 + R_L)\right|$$

Si se desprecia el valor de r_{e1} se tendrá lo siguiente:

$$R_3 < 307.4\Omega$$

Según lo anterior se escoge los valores de R_3 y R_5, iguales la mitad de los valores encontrados anteriormente para que la corriente que ingrese a las bases de T_{R1} y T_{R2} sea la mitad de la corriente que circula por las resistencias R_3 y R_5, respectivamente. Estos valores son los siguientes:

$$R_3 = 150\,\Omega$$
$$R_5 = 150\,\Omega$$

Se escoge como T_{R3} y T_{R4} a los elementos ECG291 y ECG292 respectivamente. Las corrientes que atraviesan por R_3 y R_5 se las encuentra de la siguiente manera:

$$I_{ETR3} = I_{BTR1} + I_{R3} = 0.497\text{mA}$$

En donde $I_{R3} = 2I_{BTR1}$ con lo que se tiene que:

$$I_{BTR1} = \frac{I_{ETR3}}{3} = \frac{0.497\text{mA}}{3} = 0.165\text{mA}$$

$$I_{R3} = \frac{2\ \ I_{ETR3}}{3} = \frac{2\ \ 0.497\text{mA}}{3} = 0.33\text{mA}$$

Con estos datos se calcula el valor del voltaje que cae en R_1

$$I_{ETR3} = \beta \cdot I_{BTR1} = 70 \cdot 0.165\text{mA} = 11.55\text{mA}$$
$$V_{R1} = R_1 \cdot I_{ETR3}$$
$$V_{R1} = 0.33 \cdot 0.01155 = 3.81\text{mV}$$

Como se conoce que $I_{ETR4} \approx I_{CTR4}$, se tiene que en el punto 3 se tendrá corrientes diferentes cada medio ciclo ($I_{R3} \neq I_{CTR4}$), por lo que es necesario colocar una resistencia (R_4), que limite la corriente en el colector de T_{R4}. La condición que debe cumplirse para encontrar el valor de R_4 es la siguiente:

$$I_{R3} = I_{R4} \quad \Rightarrow \quad I_{R4} = 0.33\,\text{mA}$$

Se asume que el voltaje sobre R_4 es muy pequeño para que no haya perdida de señal. Para un $V_{R4} = 1\,mV$ se tiene el siguiente valor de R_4:

$$R_4 = \frac{V_{R4}}{I_{R4}} = \frac{1mV}{0.33mA} = 3.07\Omega$$
$$R_4 = 10\Omega$$

- **Etapa excitadora:**

Esta etapa es la que proporciona la ganancia de voltaje al conjunto y para realizar este trabajo se ha escogido nuevamente al elemento ECG291.

Para encontrar la resistencia de colector R_C, $(R_C = R_6 + R_7)$ de T_{R5}, debe cumplirse la siguiente condición, para que el voltaje de alimentación V_{CC} no sea muy grande.

$$R_C < Z_{Inp}$$

En donde Z_{Inp} se calcula de la siguiente manera:

$$Z_{Inp} \approx (\beta_{TR3} + 1).\{r_{eTR3} + R_3 \,||\, [(\beta_{TR1} + 1)\ \ (r_{eTR1} + R_1 + R_L)]\}$$

Si se desprecia los valores de r_e, se tiene el siguiente valor de Z_{Inp}:

$$Z_{Inp} = (75 + 1).\{150 \,||\, [(70 + 1).(0.33\Omega + 4\Omega)]\} = 7.66\,K\Omega$$

Se asume: $Rc = 6.6\,K\Omega$

Para encontrar el voltaje en Rc se utiliza la siguiente expresión:

$$V_{Rc} \geq \frac{R_c}{R_c \,||\, Z_{Inp}} V_{omax}$$
$$V_{Rc} \geq \frac{6.6\,K\Omega}{6.6\,K\Omega \,||\, 7.66\,K\Omega} 14.14\,V$$
$$V_{Rc} \geq 26.32\,V$$

Entonces:

$$I_{RC} = \frac{V_{RC}}{R_c} = \frac{26.32V}{6.6K\Omega} = 3.99mA$$

Con los datos dados se puede calcular el valor del voltaje de la fuente de alimentación (V_{CC}), con la siguiente expresión:

$$V_{cc} \geqslant V_{RC} + V_{BE_{TR3}} + V_{R3}$$
$$V_{cc} \geqslant 26.32\,V + 0.7\,V + 0.049\,V = 27.07\,V$$
$$V_{CC} = 28V$$

En el colector de T_{R5} se forma un circuito de polarización para T_{R3} y T_{R4}, este circuito está conformado por dos diodos, los cuales serán los encargados de mantener los voltajes de base-emisor de cada transistor a 0.7 Voltios. Para asegurar el voltaje de polarización de los diodos se coloca en serie una resistencia (R_8).

Para encontrar el valor de R_8 se toma en cuenta el voltaje que debe proporcionar el circuito de polarizacion (V_p):

$$V_p = V_{BETR3} + V_{R3} + V_{R4} + V_{BETR4}$$
$$V_p = 0.7V + 0.0495V + 0.003V + 0.7V = 1.4528V$$

Con este dato y asumiendo que la corriente que ingresa a la base de TR_3 es despreciable en comparación a la corriente que circula por el circuito de polarizacion ($I_{RC} = I_p$) se tiene que la resistencia del circuito de polarizacion es la siguiente:

$$R_p = \frac{V_p}{I_p}$$

$$R_p = \frac{1.4528V}{3.99mA} = 364.25\Omega$$

Esta resistencia de polarización es igual a:

$$R_p = R_{D1} + R_{D2} + R_8$$

R_D es la resistencia estatica del diodo y se la encuentra con la siguiente expresion:

$$R_D = \frac{V_D}{I_D}$$

En donde V_D es el voltaje del diodo en su punto de operación (0.7 V) e I_D es la corriente que atraviesa por el diodo ($I_D = I_{RC}$), con esto se tiene que la resistencia estatica del diodo es igual a:

$$R_D = \frac{0.7V}{3.99mA} = 175.43\Omega$$

Con estos datos se puede encontrar el valor de la resistencia R_8:

$$R_8 = R_p - R_{D1} - R_{D2}$$
$$R_8 = 364.25\Omega - 175.43\Omega - 175.43\Omega = 13.38\Omega$$
$$R_8 = 10\Omega$$

Para mejorar la estabilidad térmica del circuito excitador se coloca en el emisor de T_{R5}, la resistencia R_9, para encontrar el valor de R_9 se encuentra la corriente de emisor y se asume que el voltaje que cae en esta resistencia es de 0.5V.

$$I_{ETR5} \approx I_{Rc} = 3.99mA$$

$$R_9 = \frac{V_{R7}}{I_{ETR5}} = \frac{0.5V}{3.99mA} = 125.31\Omega$$

$$R_9 = 150\Omega$$

Para que R_9 no disminuya la ganancia de voltaje de la etapa excitadora se conecta en paralelo a ésta un capacitor C_1, el cual debe tener una reactancia X_{C1} mucho menor que R_9, de tal manera que para una señal alterna ésta circule por X_{C1} y no por R_9, con esto se logra que R_9 no afecte la ganancia de esta etapa. Entonces un valor para C_1 seria el siguiente:

$$X_{C1} = \frac{1}{2 \cdot \pi \cdot f \cdot C_1} \leq 150$$

$$C_1 \geq \frac{1}{2 \cdot \pi \cdot f \cdot 150}$$

$$C_1 = 100\ \mu F$$

- **Etapa de Entrada**

Para la etapa de entrada se utiliza un amplificador diferencial por las ventajas antes vistas. Para conformar esta etapa se ha elegido al elemento ECG 290A.

Elemento	Tipo	V_{CBo} (V)	V_{CEo} (V)	V_{EBo} (V)	I_{Cmax} (A)	P_{Dmax} (W)	F_t (MHz)	β
ECG 290A	PNP	100	80	5	0.5	0.5	120	100

TABLA 1.6) Características de los transistores del amplificador diferencial

Los datos que se tiene para diseñar esta etapa es la corriente de Base de T_{R5} (I_{BTR5}) y el voltaje que cae sobre la resistencia R_{15}. Para encontrar el valor de

la resistencia R_{15}, se asume que la corriente que atraviesa por R_{15} es mayor que I_{BTR5}, de esta forma se tiene lo siguiente:

$$I_{BTR5} = \frac{I_{RC}}{\beta_{TR5}+1} = \frac{3.99mA}{100+1} = 39.5\mu A$$
$$I_{R15} > I_{BTR5}$$
$$I_{R15} = 10.I_{BTR5}$$
$$I_{R15} = 0.4mA$$

El voltaje sobre R_{15} se lo obtiene de la siguiente forma:

$$V_{R15} = V_{BE_{TR5}} + V_{R9}$$
$$V_{R15} = 0.7V + 0.59V$$
$$V_{R15} = 1.3V$$

Con lo que se tiene que R_{15} es igual a:

$$R_{15} = \frac{V_{R15}}{I_{R15}} = \frac{1.3V}{0.4mA} = 3250\Omega$$
$$R_{15} = 3300\ \Omega$$

Para que el amplificador diferencial opere adecuadamente se debe tener que $R_{15} = R_{14}$, entonces:

$$R_{14} = 3300\ \Omega$$

Para dar estabilidad térmica al circuito amplificador diferencial y aumentar la impedancia de entrada se colocan las resistencias R_{12} y R_{13} en los emisores de T_{R6} y T_{R7}. A estás resistencias se las escoge de valores muy pequeños para que no disminuyan la ganancia del amplificador diferencial. Por lo cual se tendrá que:

$$R_{12} = 100\ \Omega$$
$$R_{13} = 100\ \Omega$$

La resistencia R_{11} viene a ser la resistencia de emisor del amplificador diferencial, y cuyo valor debe ser muy elevado, de tal manera que se reduzca la ganancia en modo común y se obtenga tener poca distorsión.

Para dimensionar esta resistencia se asume que sobre esta resistencia cae aproximadamente un voltaje igual a $V_{cc} - V_{EB}$ (tomando como despreciable a V_{R12}), entonces el voltaje en R_{11} es 27.3V. La corriente que circula por esta resistencia es igual a la suma de las corrientes de los emisores de T_{R6} y T_{R7}, es decir :

$$I_{R12} = I_{R13} = I_{R15} + I_{BTR5} = 0.4mA + 40\mu A = 0.44mA$$

$$I_{R11} = I_{R12} + I_{R13} = 0.88mA$$

Entonces:

$$R_{11} = \frac{V_{R11}}{I_{R11}} = \frac{27.3\,V}{0.88\,mA} = 31.02\ K\Omega$$

$$R_{11} = 33\ K\Omega$$

La impedancia de entrada de esta etapa se la calcula con la siguiente expresión:

$$Z_{in_D} = 2(\beta_{TR7} + 1)[(r_{e_{TR7}} + R_{12})$$

En donde se asume que:

$$r_{e_{TR6}} = r_{e_{TR7}} = \frac{26\,mV}{I_{R12}} = \frac{26\,mV}{0.44\,mA} = 59.1\Omega$$

$$\beta_{TR7} = 100$$

Con lo que se obtiene el siguiente valor de la impedancia:

$$Z_{in_D} = 32.138\ k\Omega$$

- Etapa de Realimentación

Se utiliza la etapa de realimentación por las ventajas antes descritas, la realimentación se logra mediante las resistencias R_f y R_h, las cuales forman un divisor de voltaje entregando a la base del T_{R6} una porción de la señal de salida.

Para poder dimensionar a los elementos de la etapa realimentadora es necesario conocer la ganancia del amplificador sin realimentación:

Ganancia del Amplificador diferencial:

$$Av_D = \frac{R_c}{2.r_{e_{TR7}}}$$

En donde:

$$r_{eTR7} = \frac{26mV}{I_{ETR7}} = \frac{26mV}{0.44mA} = 59.1\Omega$$

$$Av_D = \frac{3300\Omega}{2 \cdot 59.1\Omega} = 27.92$$

Ganancia de la etapa excitadora:

$$Av_E = \frac{R_7 \,||\, [(\beta_{TR3} + 1)(r_{eTR3} + R_3 \,||\, ((\beta_{TR1} + 1)(r_{eTR1} + R_1 + R_L)))]}{r_{eTR5}}$$

En donde:

$$r_{eTR5} = \frac{26mV}{I_{ETR5}} = \frac{26mV}{3.99mA} = 6.51\Omega$$

Asumiendo a r_{eTR3} y a r_{eTR1} despreciables se tiene:

$$Av_E = \frac{3300 \,||\, [(75 + 1)(150 \,||\, ((70 + 1)(0.47 + 4)))]}{6.51} = 355.4$$

Entonces la ganancia de voltaje del amplificador sin realimentación es igual a:

$$A_v = Av_D \cdot Av_E$$
$$A_v = 27.92 \times 355.4 = 9923$$

La ganancia del amplificador con la red de realimentación se define con la siguiente expresión:

$$Av_f = \frac{A_v}{1 + B_v \cdot A_v}$$

Para el caso $B_v.A_v >> 1$, la expresión anterior se reduce a la siguiente forma:

$$Av_f = \frac{1}{B_v}$$

En donde B_v es la ganancia de la red realimentadora y su valor es el siguiente:

$$B_v = \frac{R_h}{R_h + R_f}$$

Si se desea que el amplificador de potencia tenga una ganancia total de 70 (36.9 dB), de tal manera que la sensibilidad de entrada para una carga de 4 Ω, sea menor que 1 voltio pico, entonces el factor B_v tiene que ser igual a 1.4×10^{-2}.

Para obtener este factor se puede dar los siguientes valores para R_f y R_h:

$$R_f = 39\ k\Omega$$
$$R_h = 560\ \Omega$$

Por lo tanto la ganancia del amplificador de potencia con realimentación es igual a:

$$Av_f = 70.64$$

- **Etapa de protección**

La inclusión del circuito de protección es optativa, debido a que no forma parte de la funcionalidad del amplificador; ya sea si se arma o no sobre la placa, el amplificador funcionará igualmente bien. Sin embargo es altamente recomendable su inclusión por la seguridad que presenta. Esta etapa cumple la función de salvaguardar la integridad de los transistores de salida T_{R1} y T_{R2} y a los excitadores T_{R3} y T_{R4} contra cortocircuitos en la conexión de la carga o sobre tensiones.

El circuito que muestra este tipo de protección es el siguiente:

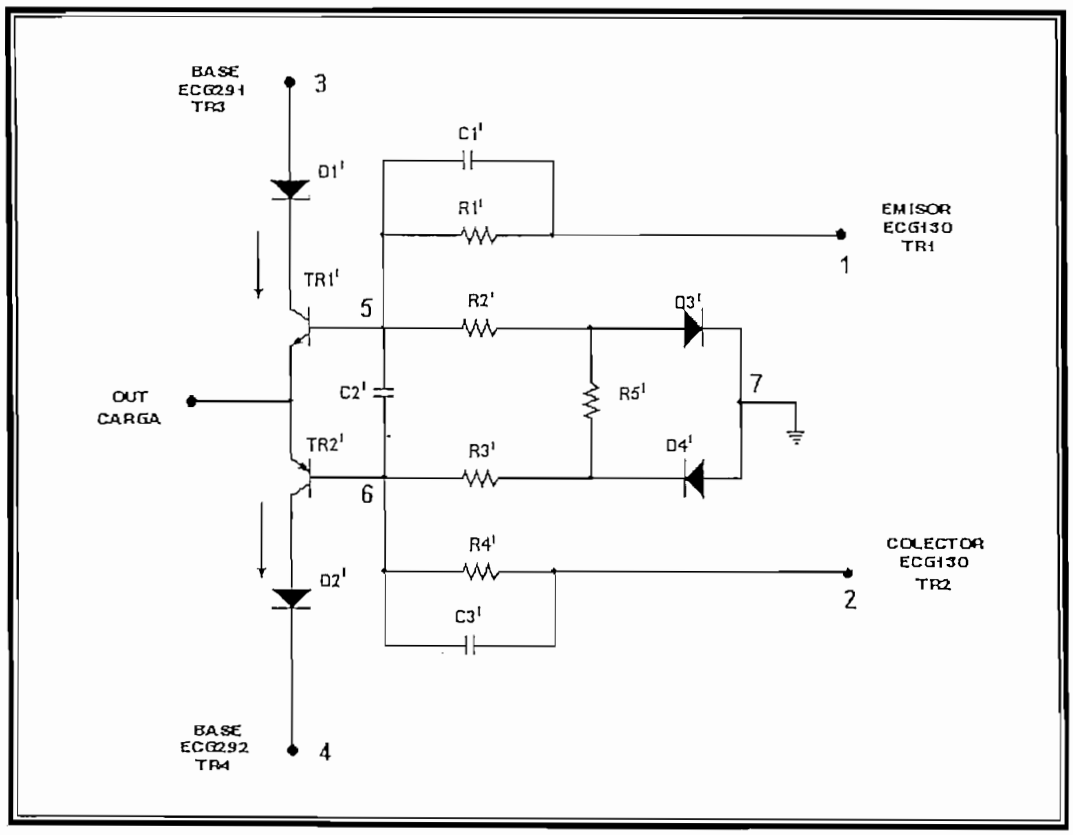


FIGURA 1.55) Circuito de protección contra cortocircuito o sobre tensiones en la carga

La operación de este circuito es la siguiente:

El circuito de protección censa por los terminales 1 y 2 el voltaje existente en las resistencias R_1 y R_2 (figura 1.54) respectivamente, al producirse un cortocircuito el voltaje en estas resistencias se eleva, el circuito de protección detecta automáticamente este cambio y satura a los transistores $T_{R1'}$ y $T_{R2'}$ de la protección.

Al ponerse $T_{R1'}$ y $T_{R2'}$ en saturación obliga a la señal a circular por estos hacia tierra y no hacia las bases de T_{R3} y T_{R4}, evitándose la circulación de corrientes y que ni los transistores de salida como los de excitación sufran algún daño.

Para llevar a los transistores $T_{R1'}$ y $T_{R2'}$ a saturación se disponen las resistencias $R_{1'}$, $R_{2'}$, $R_{3'}$, $R_{4'}$ y $R_{5'}$ de polarización respectivamente; para el caso de $T_{R1'}$ se forma un divisor de voltaje con las resistencias $R_{1'}$ y $R_{2'}$, en donde el voltaje de $R_{2'}$ es el que se alimenta a la base de $T_{R1'}$ y es el que provoca la saturación de $T_{R1'}$. Para el caso de $T_{R2'}$ es exactamente igual.

Los diodos $D_{3'}$ y $D_{4'}$ se los utiliza para evitar que ingrese alguna señal por la referencia (punto 7) al momento de existir un cortocircuito.

Los diodos $D_{1'}$ y $D_{2'}$ se los coloca para restringir el paso de la señal de acuerdo a su sentido de flujo, lo permitido es que la señal fluya como muestran las flechas de la figura anterior, lo contrario provocaría una distorsión en la señal de salida.

La característica principal de los transistores $T_{R1'}$ y $T_{R2'}$ es su alta velocidad de conmutación.

Los capacitores $C_{1'}$ y $C_{2'}$ aceleran la conducción de los transistores de protección cuando sea el caso. El capacitor $C_{3'}$ se lo utiliza para acoplar la señal

en los puntos 5 y 6 para el caso en que $T_{R1'}$ y $T_{R2'}$ deban conducir casi simultáneamente, acelerando su conmutación para señales de alta frecuencia.

- **Elementos Adicionales**

C_4 es un capacitor de desacoplo el cual minimiza la tension de offset a la salida. La reactancia presentada por este capacitor debe ser baja para que no afecte la ganancia del amplificador a altas frecuencias, un valor adecuado para este capacitor es el siguiente:

$$C_4 = 10\ \mu F$$

Para conseguir una máxima simetría en la señal de salida, a la resistencia R_C se la divide en dos partes $(R_6 + R_7)$ y en cuyo punto de unión se coloca un capacitor C_3 que va conectado a la etapa realimentadora, con la finalidad de tomar una porción de la señal de salida para equilibrar las señales en los puntos 1 y 2 (figura 1.54). La reactancia de este capacitor a una frecuencia de 20 Hz debe ser despreciable con respecto a R_6 $(10.X_{C3} = R_6/10 = 330\ \Omega)$, para que se logre lo requerido. Para calcular el valor del capacitor se aplica la siguiente expresión:

$$C_3 = \frac{1}{2 \cdot \pi \cdot 20Hz \cdot 330\Omega} = 24.11\mu F$$

$$C_3 = 47\ \mu F$$

$$R_6 = 3.3\ k\Omega$$

$$R_7 = 3.3\ k\Omega$$

La carga R_L que se conecta a la salida no es puramente resistiva conforme aumenta la frecuencia, es decir mientras aumenta la frecuencia, el parlante que esta compuesto por una bobina, ingresa una reactancia inductiva al valor de la carga por lo que esta aumenta provocando que disminuya el nivel de corriente sobre esta y por ende disminuya la potencia.

Ahora, lo que se utiliza para resolver este inconveniente es conectar en paralelo un capacitor en serie con una resistencia, a lo que se le conoce como red de Zobel. El efecto del capacitor de esta red hace que elimine la reactancia inductiva del parlante para altas frecuencias, para tratar de dar a la carga solamente una componente resistiva.

Existe un sitio WEB[5] en el cual existe un software calculador de redes Zobel (entre otras utilidades). Este software entrega los valores de la resistencia y el capacitor que compondrán esta red, la información que requiere este programa es la resistencia y la inductancia del parlante a conectar[6].

El valor de inductancia que usualmente tienen los parlantes (no profesionales) es de 10 μH, con lo que la red de Zobel para una componente resistiva de 4 Ω estaría compuesta por los siguientes elementos (según el software antes mencionado):

$$R_z = 5\,\Omega$$
$$C_z = 0.4\,\mu F$$

El capacitor C_2 trabaja como un circuito realimentador, utilizado para eliminar cualquier tendencia a la inestabilidad para altas frecuencias.

Para proteger a la etapa de entada del amplificador debe tomarse en cuenta la corriente y el voltaje de la señal de entrada. Para limitar el nivel de voltaje se coloca la resistencia R_{17} cuyo valor debe ser pequeño en comparación con la impedancia de entrada del amplificador diferencial para que el voltaje que caiga en esta resistencia sea despreciable en comparación con el que ingresa al amplificador.

---

[5] http://www.mcmantom.com/SP/utility/

[6] Para un análisis teorico acerca de las redes de Zobel remitirse a la siguiente fuente: Introduction to Electroacoustics and Audio Amplifier Desig, Second Edition, Revised Printing, by W. Marshall Leach, Jr published by Kendal/Hunt, 2001.

$$R_{17} = 1\,k\Omega$$

Para limitar el nivel de corriente de la señal que ingresa, se coloca una resistencia R_{16} en paralelo a la impedancia de entrada del amplificador, el valor de la resistencia debería ser mayor al de la impedancia de entrada, sin embargo se utiliza valores iguales para que el amplificador no tenga que hacer mucho. Esta resistencia evacua cualquier exceso de corriente que existiere.

$$R_f = 39\,k\Omega$$

Adicionalmente se coloca en paralelo a esta resistencia un capacitor C_6 para que filtre cualquier señal parásita que se pueda adquirir. El valor de este capacitor debe ser muy bajo para que no disminuya a la señal de entrada.

$$C_6 = 470\,pF$$

También es necesario conectar a la entrada un capacitor que filtre la componente DC, cuya reactancia debe de ser mucho menor que la impedancia de entrada del amplificador realimentado, por lo que el valor de C_7 para una frecuencia de 20 Hz podría ser el siguiente:

$$C_7 = 10\,\mu F$$

Todo el diseño antes descrito es idéntico para el canal 2.

El circuito amplificador de audio antes diseñado, incluido la etapa de protección se muestra en la figura 1.56.

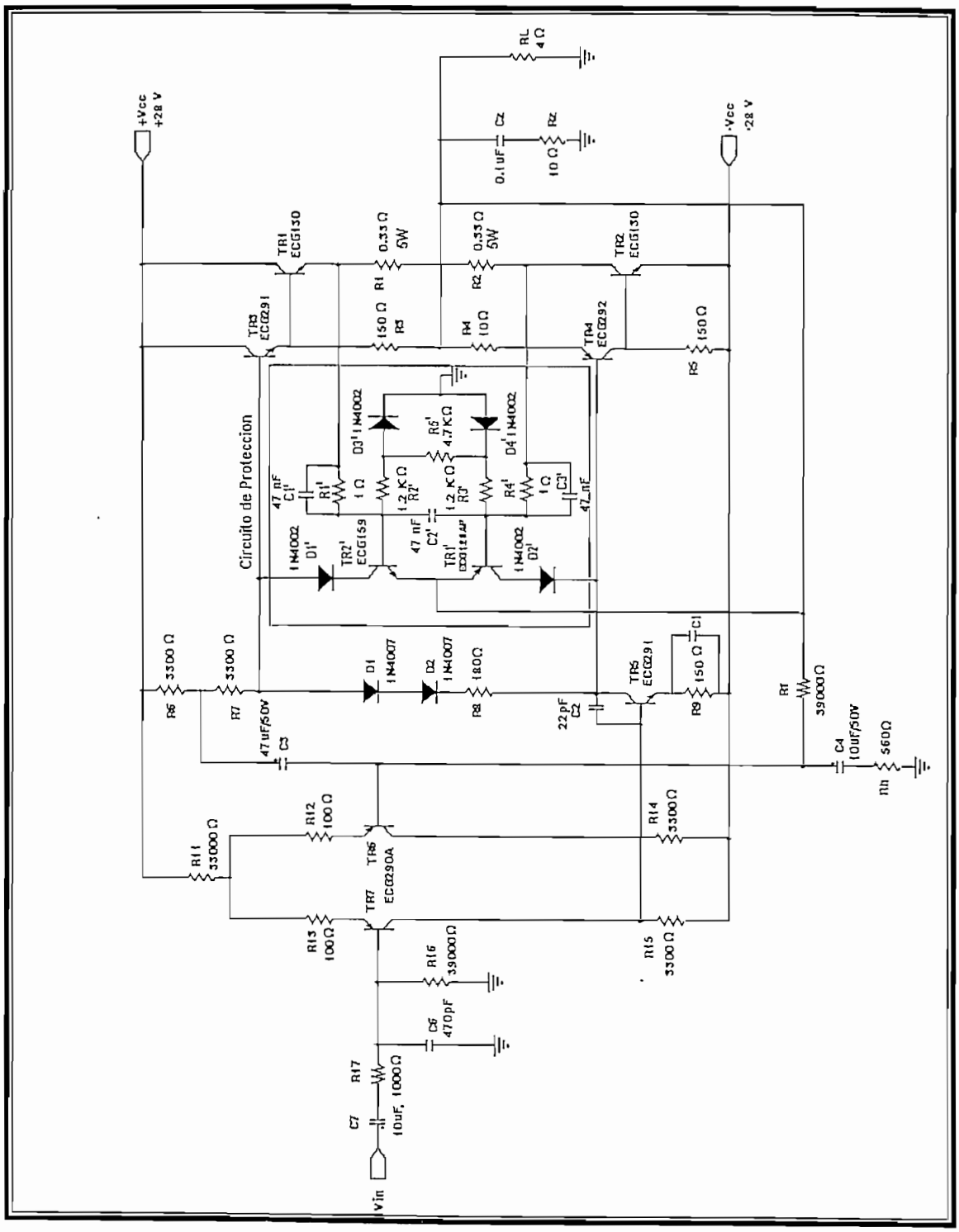


FIGURA 1.56) Circuito Amplificador de Audio diseñado (para un canal)

1.11.3 DISEÑO DEL MEZCLADOR

De cada preamplificador se obtiene una señal, que se la puede variar de acuerdo a la manipulación del potenciómetro de salida, de tal manera que es capaz de manejar las señales a ser amplificadas.

Este circuito mezclador se encargará de sumar todas las entradas que ingresen a él y entregar como resultado una sola señal que ingresará a la etapa de amplificación de potencia.

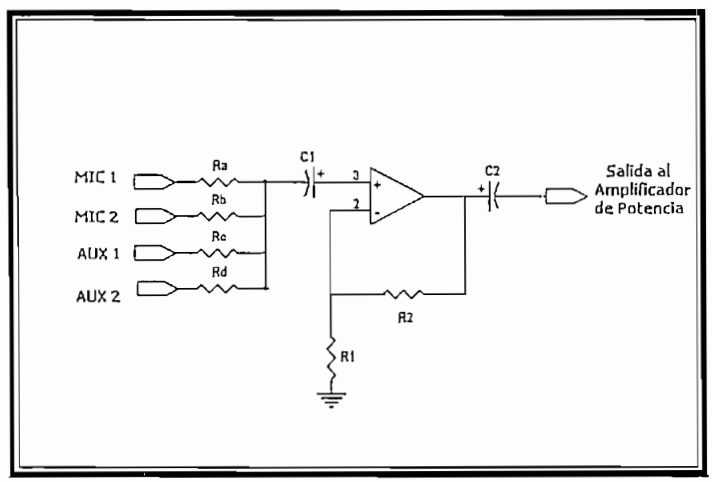


FIGURA 1.57) Circuito mezclador de cuatro entradas

En este proyecto la operación de la combinación de las señales (mezcla) se lo realizará internamente en el computador, como se verá posteriormente.

1.12 MONTAJE E IMPLEMENTACION

1.12.1 FUENTE DE ALIMENTACIÓN

Esta parte es muy importante en este sistema, ya que es la encargada de proveer de la energía necesaria para que el sistema pueda realizar el trabajo requerido. Los requerimientos de la fuente a implementarse son que sea capaz de

entregar ±27 voltios con un amperaje de máximo 3,5 amperios y ±18 voltios con un amperaje de máximo 1 amperio.

El diagrama de la fuente implementada que cumple con las indicaciones antes dadas es el siguiente:

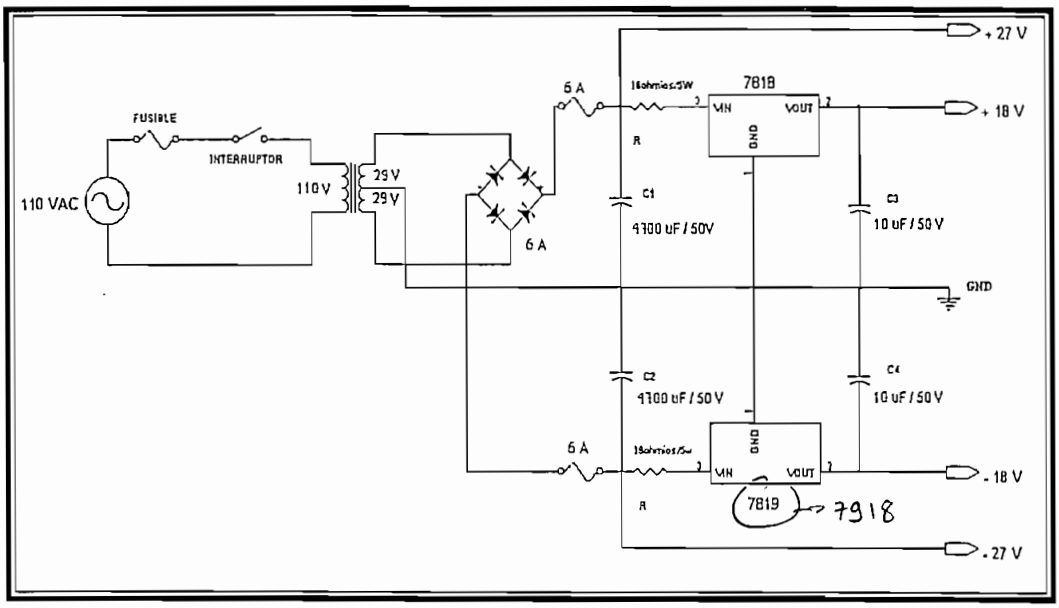


FIGURA 1.58) Fuente de alimentación

El valor del transformador de toma central se lo calculo mediante la expresión dada anteriormente (Ec.1.23). En el anexo A se puede encontrar la placa ruteada que se utilizó para esta implementación.

1.12.2 ETAPA DE PREAMPLIFICACIÓN

Para esta etapa es importante un CI que tenga una alta impedancia de entrada y una baja impedancia de salida, acompañado de una alta rapidez de conmutación para que no haya distorsión.

Como opciones que se analizó para la etapa de preamplificación fueron los CI's LM324 y TL084, se escogieron estos por ser encapsulados que contienen internamente 4 amplificadores operacionales que trabajan independientemente, y nos ayudaría a reducir espacio y complejidad en nuestro circuito.

Debido a que se trabaja con señales de audio se escogió el CI TL084 por las siguientes razones:

- Por ser un CI construido en base de FET's tiene una alta impedancia de entrada, que a su vez da una alta impedancia a la entrada de la etapa de preamplificación que es lo que se pretendía para que exista la máxima transferencia de energía por parte de los micrófonos y entradas auxiliares hacia la etapa preamplificadora.

- Por el mismo hecho que trabaja basado en FET's su tiempo de respuesta a cualquier variación es mucho más rápido y que se lo cuantifica mediante el parámetro slew rate (SR). Esta característica dará una mayor fidelidad en nuestro sistema de audio.

- No produce mucho ruido y consume poca potencia, igualmente por la característica de estar construido sobre la base de FET's

Mas información acerca del TL084 se muestra a continuación:

1.12.2.1 Características Principales:

- Bajo consumo de energía
- Protección frente a cortocircuito a la salida
- Baja distorsión armónica total (típico: 0.003%)
- Alta impedancia de entrada (JFET usados en el amplificador diferencial)
- Alto Slew rate (13 V/μs)
- Ancho de banda (3MHz.)

1.12.2.2 Estructura Física:

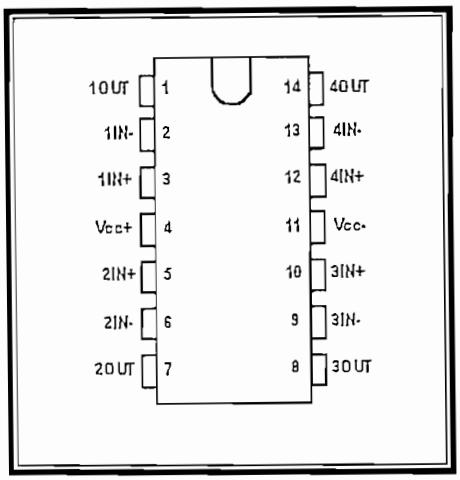


FIGURA 1.59) Diagrama de distribución de pines del TL084

1.12.2.3 Estructura Interna:

La estructura interna del circuito integrado TL084 se muestra a continuación:

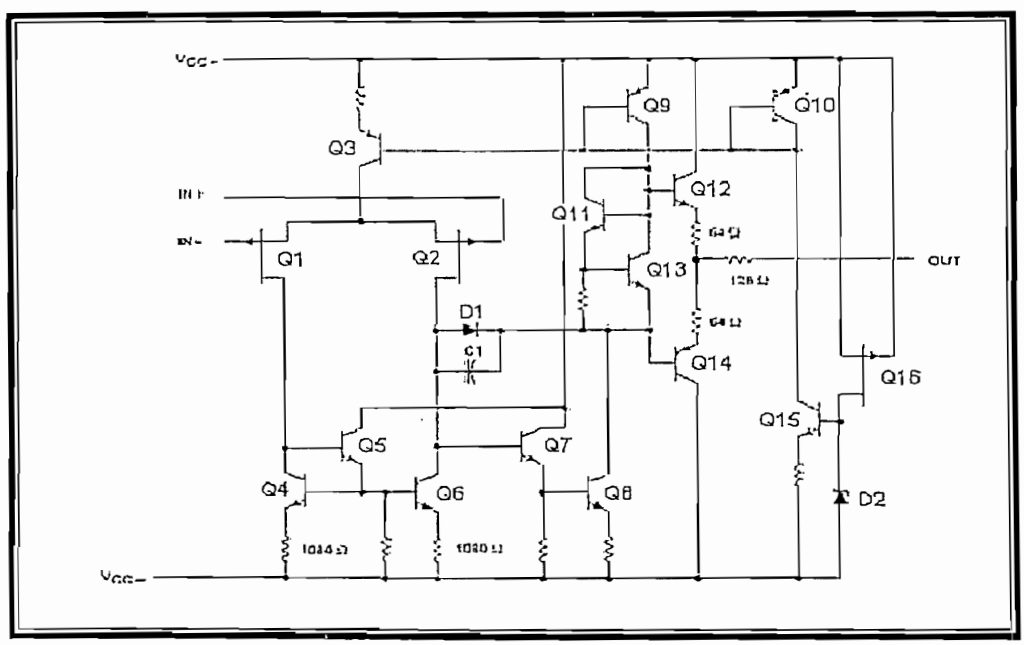


FIGURA 1.60) Estructura interna del TL084

Mas información acerca del esquemático, del ruteo y de las hojas de datos del TL084 se encuentran en el anexo C.

1.12.3 ETAPA DE POTENCIA

Para la etapa de potencia se debe tomar en cuenta la versión de encapsulado de los transistores de potencia. Encapsulado tipo TO-66 para el ECG291 y el ECG292 mientras que encapsulado tipo TO-3 para ECG130.

En el caso del ECG130 el colector es la carcaza o envoltura metálica del transistor. Estos transistores manejan toda la potencia del amplificador y deben tener un disipador que generalmente es una placa de aluminio con o sin aletas.

Los transistores deben estar en contacto con el disipador para que el calor que se introduce en su interior fluya o escape por el disipador, pero a su vez deben estar aislados eléctricamente del disipador. Para poder cumplir con ambas condiciones se usan separadores de mica y además pasta aislantes, que tiene la propiedad de conducir el calor rápidamente.

Antes de conectar los transistores al circuito se debe montarlos en el disipador, porque esto exige un trabajo y una preparación mecánica que no podría realizarse si los transistores estuvieren soldados. Sin embargo, una vez colocados es posible retirar los transistores sin sacar el disipador, porque ya están hechos todos los orificios y trabajos mecánicos de fijación del disipador.

Los demás elementos se los dispone normalmente como se muestra en el ruteado de la placa que se encuentra en el anexo A.

1.13 PRUEBAS Y RESULTADOS

1.13.1 FUENTE DE ALIMENTACIÓN

ELEMENTO	PUNTO DE MEDICIÓN	Medida	Multímetro [rms]	Multímetro [pico]	Osciloscopio [pico]
FUENTE	Primario del Transformador	Voltaje [V]	112,80	159,52	160,00
		Frecuencia [Hz]	59,99	59,99	60,00
	Secundario del Transformador (1)	Voltaje [V]	20,47	28,95	28,00
		Frecuencia [Hz]	60,01	60,01	60,00
	Secundario del Transformador (2)	Voltaje [V]	20,88	29,53	28,00
		Frecuencia [Hz]	60,03	60,03	60,00
	+ 18 Vdc	Voltaje [V]	17,92	17,92	18,00
	- 18 Vdc	Voltaje [V]	-17,93	-17,93	-18,00
	+ 27 Vdc	Voltaje [V]	27,03	27,03	28,00
	- 27 Vdc	Voltaje [V]	-27,06	-27,06	-28,00

TABLA 1.7) Valores medidos de la fuente de energía

1.13.2 ETAPA DE PREAMPLIFICACIÓN

ELEMENTO		Canal	Medida	Multímetro [rms]	Multímetro [pico]	Osciloscopio [pico]
PREAMPLIFICADOR	MIC 2	1	V_{IN} [mV]	17,50	24,75	24,00
			V_{OUT} [V]	5,25	7,43	7,40
	MIC 1	1	V_{IN1} [mV]	17,70	25,03	24,00
			V_{OUT1} [mV]	5,39	7,63	7,32
		2	V_{IN2} [mV]	17,60	24,89	24,00
			V_{OUT2} [mV]	5,10	7,21	7,20
	AUX 1	1	V_{IN1} [mV]	101,60	143,68	160,00
			V_{OUT1} [mV]	223,50	316,10	360,00
		2	V_{IN2} [mV]	105,40	149,06	160,00
			V_{OUT2} [mV]	231,90	327,90	360,00
	AUX 2	1	V_{IN1} [mV]	104,10	147,22	160,00
			V_{OUT1} [mV]	229,02	323,88	360,00
		2	V_{IN2} [mV]	103,90	146,94	160,00
			V_{OUT2} [mV]	228,58	323,26	360,00

TABLA 1.8) Valores medidos en los preamplificadores

ELEMENTO		Canal	Ganancia de Voltaje
PREAMPLIFICADOR	MIC 2	1	300,00
	MIC 1	1	304,52
		2	289,77
	AUX 1	1	2,20
		2	2,20
	AUX 2	1	2,20
		2	2,20

TABLA 1.9) Valores de Ganancia de voltaje calculados en los preamplificadores

1.13.3 ETAPA DE AMPLIFICACIÓN

PARAMETRO	Canal 1	Canal 2
Impedancia de entrada (KΩ)	38,076	39,749
Impedancia de salida (Ω)	0,088	0,088
Frecuencia inferior (Hz)	32	29
Frecuencia superior (KHz)	66	53
Ancho de banda (KHz)	65,968	52,971

TABLA 1.10) Valores calculados de diferentes parámetros del amplificador de potencia

	Canal 1			Canal 2		
	Carga 3,7 Ω	Carga 6,8 Ω	Carga 7,3 Ω	Carga 3,7 Ω	Carga 6,8 Ω	Carga 7,3 Ω
Vin (mV)	126,9	185,9	189,9	108,8	190,6	182,2
Vo (V)	9,7	13,03	13,5	9,6	13,04	13,5
Sensibilidad (mV)	126,9	185,9	189,9	108,8	190,6	182,2
Ganancia de voltaje	76,43	70,09	71,09	88,23	68,41	74,09
Potencia	25,42	24,96	24,96	24,90	25,00	24,96
Factor de amortiguamiento (FA)	42,04	77,27	82,95	42,04	77,27	82,95

TABLA 1.11) Diferentes características medidas y calculadas del amplificador de potencia

La descripción de los equipos utilizados en la toma de medidas se muestran en el capítulo cuatro.

CAPÍTULO 2:
INTERFAZ HARDWARE

2.1 INTRODUCCIÓN

En la actualidad todo tipo de sistemas tienden a ser controlados por un computador, para obtener de esta manera obtener datos de dichos sistemas, además procesar dicha información y así poder controlar el correcto funcionamiento del sistema, que en este caso es un amplificador de potencia de audio.

Para la obtención de datos del sistema se requiere de un interfaz hardware, el cual permite el intercambio de información entre el computador (datos digitales) y el amplificador de audio (información analógica).

2.2 DEFINICIÓN

La interfaz es lo que *media*, lo que facilita la comunicación, la interacción entre dos sistemas de diferente naturaleza, específicamente el ser humano y el computador.

Esto implica, además, que se trata de un sistema de traducción, ya que los dos *hablan* lenguajes diferentes: verbal o escrito (palabra) en el caso del hombre y binario en el caso del computador.

La Interfaz, es todo aquello con lo que se interactua para lograr efectuar alguna acción en un sistema. Por ejemplo, los pedales y la palanca de cambios son parte de la interfaz de usuario del sistema de propulsión de un automóvil. El tablero es la interfaz de usuario de los sistemas de mantenimiento de cabina (temperatura, sonido, etc.)

A pesar de ser en esencia un intermediario, la interfaz no es un sistema que se ubica y opera "entre" el computador y el hombre sino que ha sido concebido como parte integrada a la propia máquina, a través de la creación de los

"periféricos" como el teclado, el ratón y la pantalla de video, y a través de los componentes gráficos que aparecen en la pantalla.

Mientras existen periféricos comunes a numerosas aplicaciones, en consecuencia, es necesario distinguir dos tipos de interfaces:

Una interfaz hardware, a nivel de los dispositivos utilizados para ingresar, procesar y entregar los datos: Teclado, ratón, monitor, tarjetas de Adquisición de Datos, etc.; y una interfaz software, destinada a entregar información acerca de los procesos y herramientas de control, a través de lo que el usuario observa habitualmente en la pantalla.

Es decir, la interfaz hardware es el intermediario entre la computadora y periféricos, o sea lo que permite la comunicación entre la computadora y su entorno, en el caso de este Proyecto de Titulación la comunicación con el amplificador de potencia de audio, esta comunicación necesita de hardware y de software; en este capítulo se describe el hardware necesario para realizar dicha comunicación.

2.3 DESCRIPCIÓN

Un sistema de adquisición de datos toma información que se obtiene de fenómenos físicos y mediante transductores los convierte en señales eléctricas para poder manipularlos en un computador; y luego mediante otros transductores nuevamente convertirlos en fenómenos físicos. Este proceso se lo puede representar mediante un diagrama de bloques como el que se muestra a continuación (Figura 2.1):

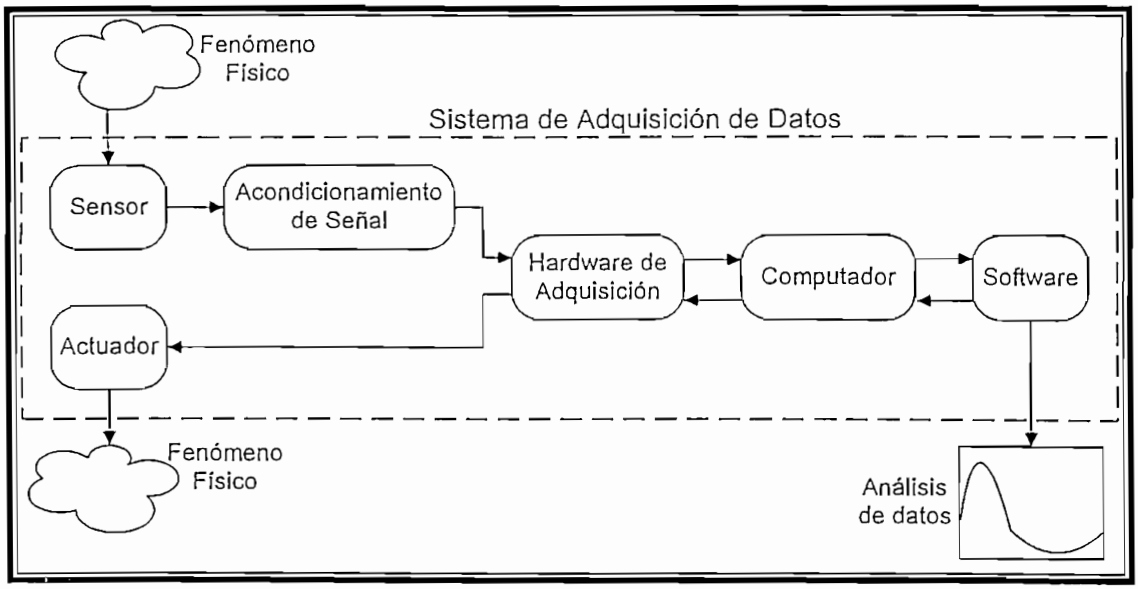


FIGURA 2.1) Diagrama de bloques de un sistema de Adquisición de datos

2.3.1 PROCESO DE ADQUISICIÓN DE DATOS

A continuación se explica la función que realizan cada uno de los elementos que forman parte en la adquisición de datos:

2.3.1.1 Sensor / Actuador (Transductores)

Sensor: convierte un fenómeno físico en un fenómeno eléctrico.

Actuador: convierte un fenómeno eléctrico en un fenómeno físico.

2.3.1.2 Acondicionamiento de Señal

Los sensores son incompatibles con el hardware de adquisición de datos, la señal entregada por los sensores necesita ser acondicionada. El tipo de acondicionamiento de señal depende del sensor que se esté utilizando, por ejemplo, una señal de muy poca amplitud necesita ser amplificada, o si una señal contiene componentes de frecuencia que no se usan en un determinado estudio o determinada aplicación se necesita filtrar, en especial se necesita eliminar el

ruido. En el proceso de acondicionar una señal se puede dar amplificación, filtrado, multiplexación, aislamiento o fuentes de excitación.

2.3.1.2.1 Amplificación

Señales de bajo nivel (alrededor de 100 milivoltios) necesitan ser amplificadas, así también señales de alto nivel pueden necesitar ser amplificadas estos depende del rango de entrada del subsistema analógico.

2.3.1.2.2 Filtrado

Al filtrar una señal se necesita eliminar el ruido que se pudo haber introducido en alguna parte del proceso de conversión del fenómeno físico en fenómeno eléctrico.

2.3.1.2.3 Aislamiento

Si la señal de interés contiene transitorios de muy alto voltaje puede dañar el computador, entonces el sensor debería ser aislado eléctricamente del computador para propósitos de seguridad.

2.3.1.2.4 Multiplexación

Es una técnica que se usa cuando se requiere adquirir varias señales con un dispositivo de una sola entrada. Primero toma una muestra de la primera entrada y conmuta a la segunda entrada, toma la muestra y conmuta nuevamente a la siguiente entrada y así sucesivamente hasta hacer un muestreo todas las señales, para luego repetir el proceso. La velocidad de muestreo de cada canal es igual a la velocidad de muestreo del conversor A/D dividido para el número de señales de entrada (canales) muestreadas.

2.3.1.2.5 Fuentes de Excitación

Algunos sensores requieren de una fuente de excitación para poder operar, por ejemplo dispositivos de resistencia variable en función de la temperatura (RTD) requieren de un voltaje externo o una corriente de excitación. El módulo de acondicionamiento de señal provee la corriente de excitación necesaria para el correcto funcionamiento de las RTD's. Por ejemplo los micrófonos de condensador necesitan una alimentación de 48 Voltios (fuente de excitación) para poder funcionar.

2.3.1.3 Hardware de Adquisición de Datos

El hardware de Adquisición de datos se caracteriza por tener subsistemas. Cada subsistema es un componente de la tarjeta de adquisición de datos que cumple una función específica. Los subsistemas son (figura 2.2):

- Entrada Analógica
- Salida Analógica
- Entrada / Salida Digital
- Contador / Temporizador

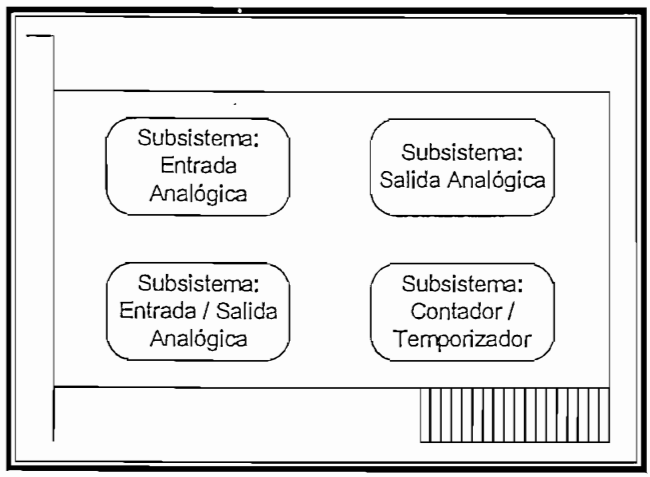


FIGURA 2.2) Diagrama de bloques de los subsistemas

2.3.1.3.1 Entrada Analógica

Los subsistemas entrada analógica (AI: Analog Input) convierten señales continuas obtenidas por el sensor a bits (señales discretas) para que puedan ser entendidas por el computador, típicamente son dispositivos multicanal que tiene 12 o 16 bits de resolución.

2.3.1.3.2 Salida Analógica

Los subsistemas salida analógica (AO: Analog Output) convierten datos digitales almacenados en el computador en señales analógicas, estos sistemas realizan un proceso inverso al realizado por los subsistemas entrada analógica, típicamente las tarjetas de adquisición de datos cuentan con dos canales de salida con 12 bits de resolución.

2.3.1.3.3 Entrada / Salida Digital

Los subsistemas entrada / salida digital (DIO: Digital Input / Output) están diseñados para tomar o enviar valores digitales (niveles lógicos) hacia el hardware o del hardware, típicamente se los puede usar como bits (líneas) o como un puerto, el cual típicamente está conformado de 8 líneas.

2.3.1.3.4 Contador / Temporizador

Los subsistemas Contador / Temporizador (C/T: Counter / Timer) son usados para contar eventos, generar trenes de pulsos, medir frecuencia y periodo.

2.3.1.4 Computador

El computador provee un procesador, un sistema de reloj, un bus de transferencia de datos, y la memoria y espacio en disco para almacenar los datos.

El procesador controla cuan rápido los datos son aceptados por el conversor. El sistema de reloj provee la información de tiempo acerca de la adquisición de datos. Los datos son transferidos del hardware al sistema de memoria mediante el acceso dinámico de la memoria (DMA: Dynamic Memory Access). La máxima velocidad de adquisición es determinada por la arquitectura del bus de datos del computador.

2.3.1.5 Software

Sirve para poder usar el hardware, se puede enviar información al hardware y recibir información del hardware, se puede enviar información de configuración como por ejemplo velocidad de muestreo y recibir información del hardware como por ejemplo datos, mensajes de error.

Se puede diferenciar dos clases de software:

- Software Controlador
- Software Aplicación

Por ejemplo, si se usa la herramienta de Adquisición de datos[7] con la tarjeta PCI-DAS 1000 de Computer Boards, el software controlador se denomina CBI.

La relación entre el usuario, el software aplicación, el software controlador y el hardware se muestra en la siguiente figura (figura 2.3):

---

[7] MATLAB: Data Acquisition ToolBox

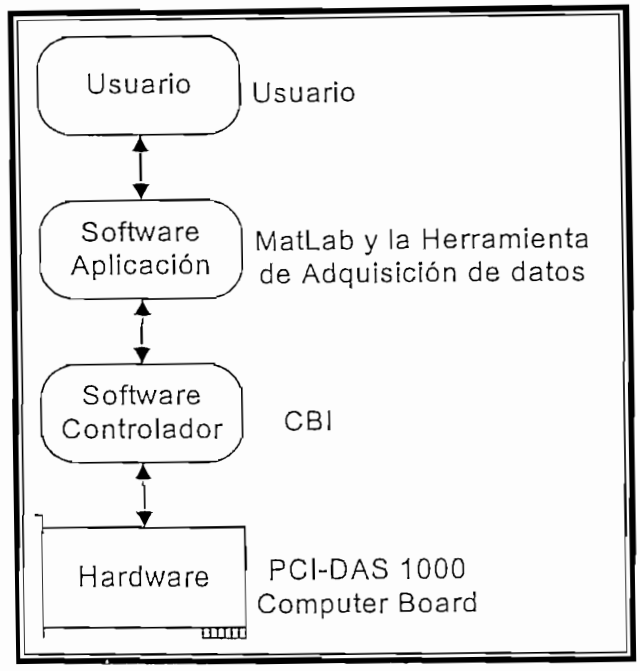


FIGURA 2.3) Relación Usuario – Software – Hardware

2.3.1.5.1 Software Controlador

Este software se usa para el dispositivo de adquisición de datos, el software controlador permite acceder y controlar las capacidades del hardware, un software controlador básico permite: Traer datos de la tarjeta, controlar la velocidad con la cual los datos son adquiridos, integrar el hardware de adquisición de datos con los recursos del computador como son interrupciones del procesador y memoria, integrar el hardware de adquisición de datos con el hardware acondicionador de señal, acceso a los diferentes subsistemas de la tarjeta de adquisición de datos, acceso a diferentes tarjetas de adquisición de datos (mismo fabricante).

2.3.1.5.2 Software de Aplicación

El software de aplicación provee el enlace entre el software controlador y el usuario, las funciones básicas del software de aplicación son: generar reportes de información como es el número de muestras adquiridas, generar eventos, gestión

de los datos almacenados en la memoria del computador, condición de la señal, graficar los datos adquiridos.

2.3.2 DESCRIPCIÓN DEL SISTEMA A IMPLEMENTARSE

En la figura siguiente (figura 2.4) se muestra un esquema de la configuración del sistema de audio:

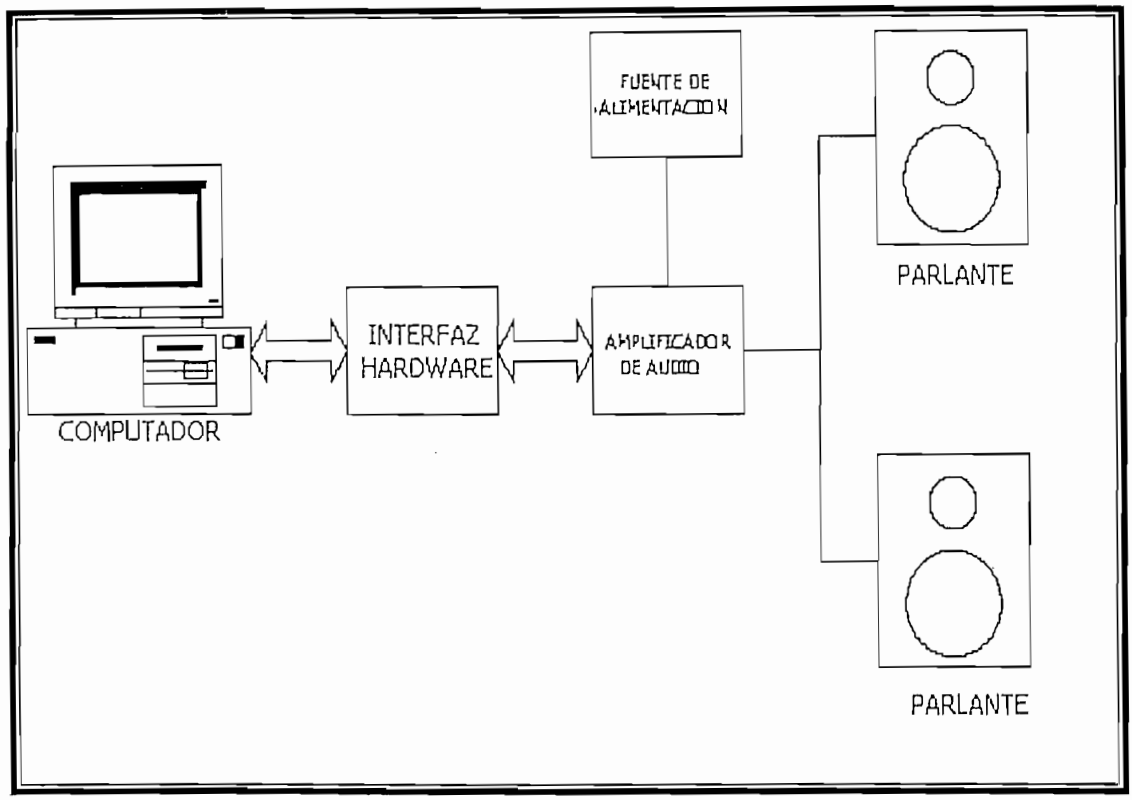


FIGURA 2.4) Diagrama de bloques del sistema a implementarse

La función del computador, es la de procesar la información que se obtiene del amplificador de potencia de audio mediante el interfaz hardware, además de almacenar el software que sirve como interfaz entre el usuario y el amplificador de potencia.

El amplificador de audio cumple la función de entregar a la carga (parlantes) la potencia necesaria; la fuente de alimentación se encarga de entregar al amplificador de potencia de audio la energía necesaria para su correcto

funcionamiento, y como se mencionó anteriormente el interfaz hardware sirve de intermediario entre el amplificador de potencia y el computador.

2.4 ESPECIFICACIÓN DE REQUERIMIENTOS

En base al análisis anteriormente realizado el interfaz hardware debe permitir:

a) Adquirir datos.

b) Enviar datos.

c) Almacenar y procesar la información.

d) Manipular valores analógicos mediante señales digitales.

2.5 DISEÑO

De acuerdo al diseño del amplificador de potencia de audio presentado en el *Capítulo 1* de este proyecto, se necesita 7 potenciómetros digitales los cuales permitan variar las ganancias de las 7 entradas de audio (1 entrada estereofónica de micrófono, 1 entrada monofónica de micrófono y 2 entradas estereofónicas de línea), los valores de estos potenciómetros están dados en el capítulo 1. A estos potenciómetros se los controla enviando pulsos mediante la tarjeta de adquisición de datos desde el computador.

Para la manipulación de los 7 potenciómetros digitales se necesita 15 salidas digitales; es decir, 1 salida digital para habilitar los potenciómetros ($\overline{CS}$) por cada preamplificador, 1 salida digital se la utilizaría para controlar el contador (contar hacia arriba "Up" o hacia abajo "Down"), esta también es por cada preamplificador y 1 salida digital para incrementar o decrementar el valor de los potenciómetros, esta es por cada potenciómetro, además se requiere una línea adicional para envío de una señal para la medición de la resistencia del parlante.

$$\text{salidas digitales} = 16$$

Además se necesita 8 entradas analógicas para realizar el muestreo de las señales, 7 para las entradas[8] (3 de micrófono y 4 de línea) y una entrada analógica para tomar el valor de la resistencia del parlante.

$$\text{entradas analógicas} = 8$$

Para el envío de la información del computador al amplificador de potencia de audio se necesita 2 salidas analógicas, esto es por ser un amplificador estereofónico.

$$\text{salidas analógicas} = 2$$

Es decir, se necesita una tarjeta de adquisición de datos que al menos tenga 16 entradas/salidas digitales, 8 entradas analógicas y 2 salidas analógicas, debido a que las tarjetas de adquisición de datos que cuentan con subsistemas de salida analógicos son mucho más costosos que las tarjetas que no cuentan con este tipo de subsistemas se aprovecha los recursos existentes en el computador como lo es la tarjeta de sonido, la cual cuenta con 2 salidas analógicas (se usan para este proyecto) y 2 entradas analógicas.

$$\text{DAQ} = \begin{matrix} 16 \text{ entradas / salidas digitales} \\ 8 \text{ entradas analógicas} \\ 2 \text{ salidas analógicas} \end{matrix}$$

[8] Las siete entradas se distribuyen de la siguiente forma: 2 para la entrada de micrófono estereo, 1 para la entrada de micrófono monofónico y 2 para cada una de las entradas auxiliares.

2.6 ELEMENTOS UTILIZADOS

a) Computador personal;

b) Tarjeta de Adquisición de Datos PCI-DAS 1000 de Measurement Computing;

c)Tarjeta de Adquisición de Datos Winsound;

d) Potenciómetros Digitales X9C503 de Xicor.

2.6.1 TARJETA DE ADQUISICIÓN DE DATOS

Las *tarjetas de adquisición de datos*, conocidas también como *daq* (data acquisition card), se conectan directamente a algún bus de datos del computador, permiten adquirir datos en tiempo real. Una tarjeta de adquisición de datos tan solo aporta los bloques de encaminamiento de la señal, así como la amplitud de ésta (con posibilidad de amplificación), las funciones de cálculo, memoria y visualización las tiene que realizar el computador al cual esté conectado la tarjeta.

Una ventaja importante en las tarjetas de adquisición de datos es el evitar la duplicidad de diferentes bloques en el instrumento y en el computador, como pueden ser memoria o funciones de cálculo. También es importante la facilidad de instalación, la puesta en funcionamiento y su flexibilidad de uso en muchas aplicaciones.

La tarjeta PCI-DAS 1000 utiliza el bus PCI (Peripheral Component Interconnect: Interconexión de Componentes Periféricos) para conectarse al computador.

Los buses actuales permiten realizar varias transferencias en una sola operación (transferencia a ráfaga).

Estas son algunas de las características de un bus:

- Banda base.- La información se envía directamente sin ningún tipo de modulación.

- Paralelismo.- Serie (bit a bit) o paralelo permitiendo el envío de toda una palabra.

- Multiplexación.- Es frecuente que las líneas de un bus se encuentren multiplexadas para permitir reducir su número. De esta manera se envían por las mismas líneas informaciones distintas, por ejemplo direcciones y datos, en diferentes instantes de tiempo.

- Temporización.- Ciclo completo en que se realiza una transferencia, depende de si se ocupa completamente el tiempo del bus para realizar esta operación. Se relaciona también con la presencia o ausencia de un único reloj que determine el inicio y fin de los ciclos, sincronismo o asincronía.

- Modo de operación.- Síncrono o asíncrono en función de la forma en que se realice el diálogo entre los dispositivos conectados.

- Estrategia de control.- En general, la CPU controla todas las transferencias entre un dispositivo maestro y el resto de los dispositivos. En los sistemas que disponen de varios procesadores hay varios maestros potenciales del bus por lo que se precisará de un mecanismo de arbitraje para que no se produzcan colisiones de datos.

- Longitud (bits).- Depende de la aplicación.

- Velocidad de transmisión.- Depende de la longitud y de la lógica de control. Velocidad = longitud * paralelismo. Se suele asociar este valor con el ancho de banda.

- Direccionamiento.- Define la forma por la que el maestro elige un destino para la transferencia. Puede ser geográfica, si las direcciones están asociadas a conectores físicos o lógica, en caso de que la información sobre la dirección esté almacenada en el dispositivo destino. En este caso de direccionamiento todos los dispositivos esclavos comparan su dirección con la enviada por el maestro; si coinciden se inicia el ciclo de transferencia de datos.

- Capacidad de conexión.- Determinada por el número de dispositivos que pueden conectarse al bus. Está determinada por el direccionamiento y las características físicas del soporte.

Características del bus PCI

Según las características antes mencionadas, el bus PCI puede definirse como un bus paralelo, multiplexado y síncrono. Estos son algunos datos más precisos sobre sus características:

- Frecuencia de operación.- 33MHz y 66 MHz.

- Velocidad de transferencia.- Típicamente 33 MHz * 32 bits = 132 MB/s. Se permite una extensión transparente de 64 bits de los buses de datos y direcciones de 32 bits, que a una frecuencia de 66 MHz darían lugar a 264 MB/s.

- Duradero.- Independencia del procesador. Idealmente se garantiza una transición de los dispositivos a futuras generaciones de procesadores.

- Configuración automática.- Mediante un espacio de configuración compuesto por una colección de registros propio de cada dispositivo. ("Plug and Play")

- Capacidad.- Soporte de varias tarjetas operando concurrentemente sobre el bus. Modo de operación "multi-master".

- Integridad de datos.- El protocolo del bus PCI proporciona control de paridad y chequeo de errores durante las transferencias de datos y direcciones.

Además del bus PCI para la conexión de tarjetas de adquisición de datos al computador (tarjeta madre) existen otras opciones de conexión como son las siguientes:

- USB (Universal Serial Bus: Comunicación Serial Universal)

Arquitectura de comunicación de datos externa, se debe contar con puerto USB. Las velocidades entre las que trabaja este tipo de bus son: 1.5 Mbps y 45 Mbps.

- RS-232 (Recommended Standard 232: Estandar Recomendado 232)

Arquitectura de comunicación de datos externa usada en PC's, Laptop's, PLC's, y otros dispositivos que soporten este protocolo (RS-232).

- ISA (Industry Standard Arquitecture: Arquitectura Estandar de la Industria)

Arquitectura de comunicación de datos interna de 16 bits.

Cada tarjeta de adquisición de datos presenta funcionalidades diferentes, lo que da la posibilidad de utilizar una tarjeta para aplicaciones muy variadas, como podría ser el contar eventos, la generación de señales de salida, o la adquisición de señales de entrada.

2.6.1.1 Características

Una tarjeta de adquisición de datos se caracteriza por una serie de parámetros que permiten decidir su utilización. Los parámetros se fijan a partir de un conjunto de funciones y dispositivos internos de la placa, entre los cuales destacan el número de canales de entrada y de salidas analógicas y digitales, los convertidores analógico-digitales y digitales-analógicos, los sistemas de multiplexación y los márgenes dinámicos de entrada y salida.

2.6.1.1.1 Entradas analógicas

El número de canales analógicos ha de distinguir entre los que permiten entrada diferencial (modo diferencial) de los de entrada simple (modo común). Las entradas simples están referenciadas a una tierra común (Figura 2.5) y se utilizan en el caso de trabajar con señales de alto nivel (tensión superior a 1v) dónde no exista grandes problemas de interferencias.

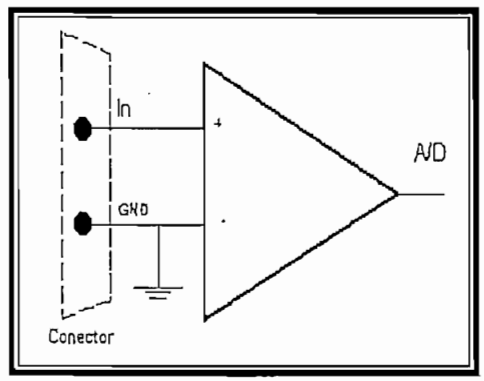


FIGURA 2.5) Canal analógico simple (modo común)

En el caso de utilizar entradas diferenciales (Figura 2.6), cada entrada tiene su propia referencia de forma que el posible ruido en modo común que se pueda introducir queda rechazado.

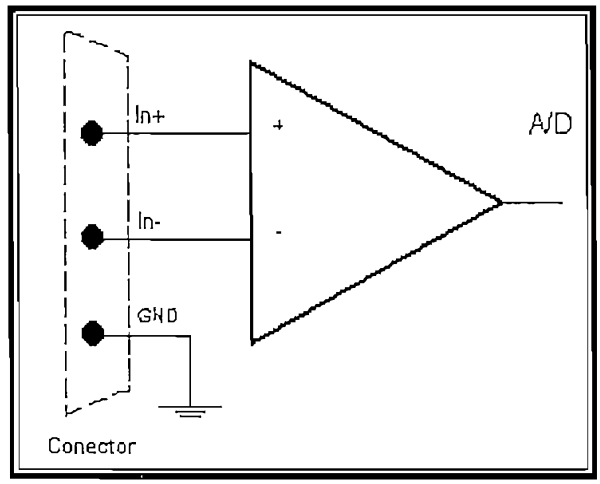


FIGURA 2.6) Canal analógico diferencial (modo diferencial)

1 Conversión analógico-digital

Las señales por su naturaleza son del tipo analógico; es decir, continuas, variando constantemente con el tiempo: y = f(t). Las señales pueden ser muy lentas como la variación de temperatura de un tanque figura 2.7) o muy rápida como una señal de audio figura 2.8).

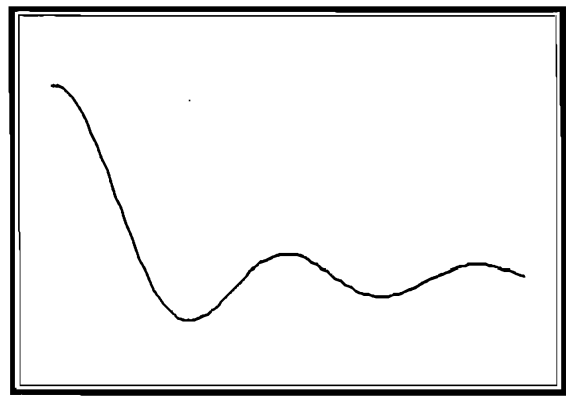


FIGURA 2.7) Onda que representa la variación de temperatura en un tanque

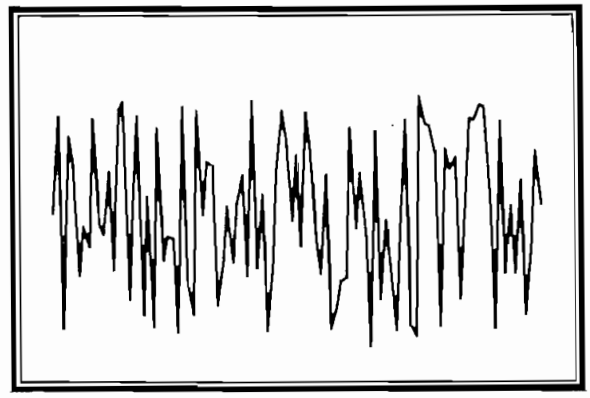


FIGURA 2.8) Onda que representa a una señal de audio

El tratamiento directo de la información analógica como comparar, realizar operaciones algebraicas, almacenar, recuperar, eliminar ruido, transmitirla a distancia, presenta cierta complejidad pero que se pueden llevar a cabo con circuitos electrónicos, sobre la base de eficientes circuitos integrados como los amplificadores operacionales, es decir; se lleva a cabo un proceso de conversión mediante un circuito electrónico denominado conversor analógico / digital (ADC). Una vez que se obtiene la información digitalizada entonces se puede realizar el tratamiento de la información por medio de microprocesadores o dentro del computador.

En muchos casos es necesario presentar los resultados finales en forma de salida analógica, por ejemplo: salida audible de música almacenada en el computador. Esto se lleva a cabo mediante un circuito electrónico denominado conversor digital / analógico (DAC).

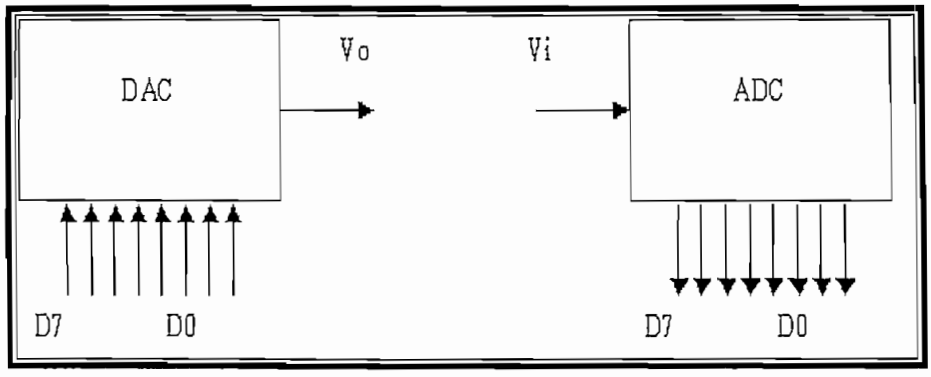


FIGURA 2.9) Proceso de conversión Digital-Análogo-Digital

La conversión analógica-digital se trata de una señal analógica que se convierte a una estructura binaria. Es decir, la conversión A/D consiste en la obtención del valor digital correspondiente al voltaje de una señal analógica de entrada, en un instante dado.

El proceso de digitalización comienza tomando muestras de la señal analógica cada cierto tiempo, la velocidad de muestreo (fm) depende de la frecuencia de la señal (f) de entrada.

La frecuencia de muestreo o velocidad de muestreo debe ser igual o mayor al doble del ancho de frecuencia de la señal original (Teorema de Nyquist). Con ello se garantiza la recuperación de la forma original.

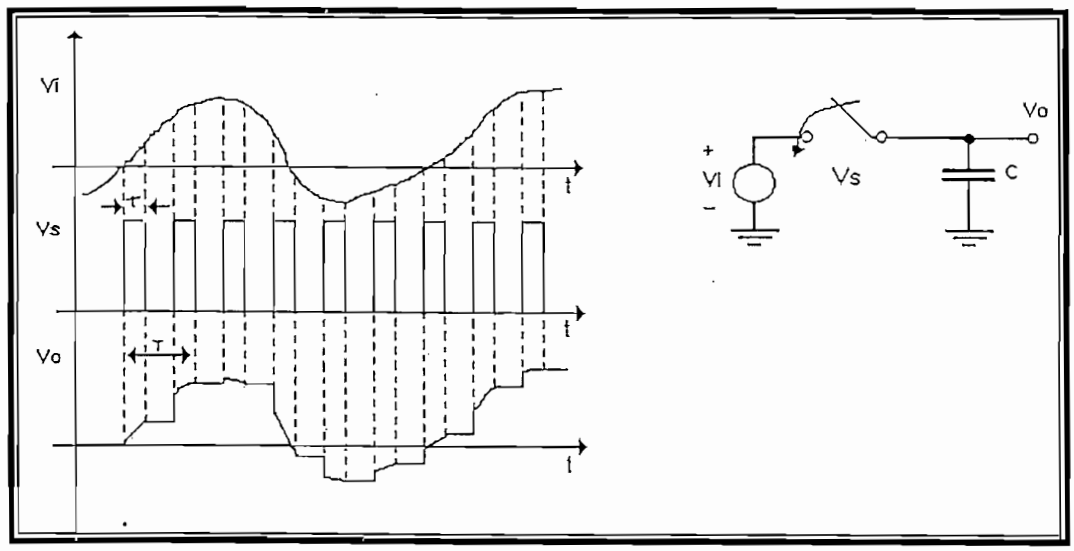


FIGURA 2.10) Proceso de digitalización de una señal analógica

La señal muestreada obtenida, no es una señal digital ya que presenta distintos niveles en cada muestra. Es necesario ahora cuantificarla y representarla por un código binario. Se puede representar con códigos de 4, 8, 10, 12 o 16 dígitos, lo que define la precisión, esto es: el número de bits que utiliza el conversor para representar la señal analógica.

Si se toma n=4 bits y el máximo nivel es 10 voltios, entonces:

$$\text{Re solución} = \frac{\text{Voltaje}}{2^n - 1}$$
$$= \frac{10}{2^4 - 1}$$
$$= \frac{10}{15}$$
$$= 0.667$$

Voltaje	Código
0.000	0000
0.667	0001
1.333	0010
1.999	0011
10.000	1111

TABLA 2.1) Relación voltaje - código

Tomando en consideración la fórmula anterior se puede deducir que el error de cuantificación se reduce aumentando el número n de dígitos.

Según el principio que se aplica se tendrá diferente velocidad de conversión (muestras/seg.); es decir, la máxima frecuencia con que pueden obtenerse nuevos resultados. Hay de distintos tipos según sean los requerimientos de velocidad y precisión: secuencial y aproximaciones sucesivas.

El conversor fija muchas de las características de la tarjeta. Cuanto mayor sea el proceso de conversión, mayores serán las posibles frecuencias de muestreo.

Las señales de entrada han de ser muestreadas según el criterio de Nyquist, por lo que es importante que el convertidor analógico-digital pueda convertir la señal en palabras digitales en el menor tiempo posible.

Un proceso rápido adquiere más valores en un tiempo dado que uno de proceso lento, esto permite representar mejor las señales originales.

La siguiente figura muestra una señal digitalizada (cuantizada) con un convertidor de tres bits que permite dividir el rango analógico en 8 intervalos.

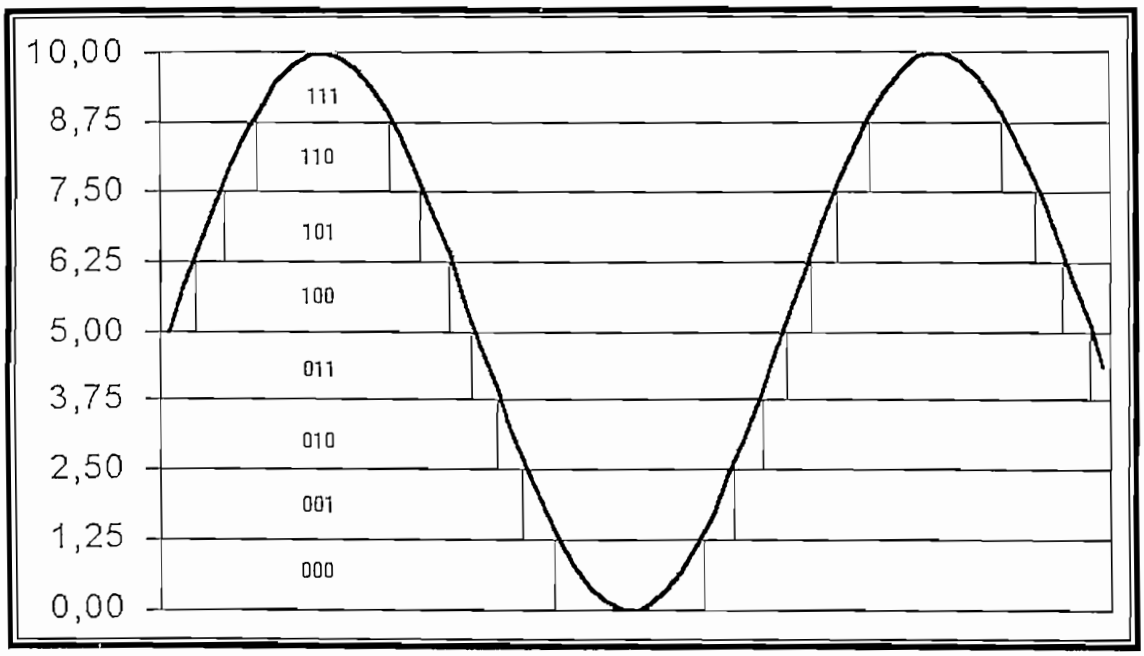


FIGURA 2.11) Señal analógica muestreada

Cada intervalo se representa mediante un código binario entre 000 y 111. A simple vista se observa que la representación digital introduce un error de cuantificación, el cual será menor cuanto mayor sea la resolución. De cualquier forma, la resolución del convertidor deberá ser suficientemente alta para detectar un mínimo cambio de tensión.

En la actualidad existen diferentes tipos de convertidores analógico-digitales. El más popular es el de aproximaciones sucesivas, ya que ofrece la máxima velocidad y resolución a un precio razonable.

El conversor análogo – digital que usa la tarjeta de adquisición de datos PCI-DAS1000 es el ADS7800, el cual es un convertidor de 12 bits que realiza la conversión mediante aproximaciones sucesivas.

Los métodos más usados para realizar la conversión analógica a digital se los detalla a continuación:

Método Secuencial

Este es un método sencillo pero lento debido a que trabaja de la siguiente manera: Se incrementa el valor digital (a la entrada del convertidor D/A) comenzando de 0 y finalizando una vez que el comparador detecta que la salida del amplificador es igual a la entrada analógica. Este valor digital será el resultado de la conversión. El problema radica en que la velocidad de conversión se alarga a medida que la entrada analógica es mas elevada; es decir, para convertir una señal equivalente a 10 (00001010) sè necesitarán 10 pasos de prueba, en cambio con una señal equivalente a 127 (01111111) se necesitarán 127 comparaciones antes de detectar el valor.

Método de Aproximaciones Sucesivas

Este es el método mas usado y veloz, aunque requiere una programación algo mas compleja. Consiste en configurar a "1" lógico cada bit comenzando por el mas significativo. Por lo tanto, si se trabaja con 8 bits de resolución, el primer paso es colocar a 1L el bit 8 (sería 1000000=128 en decimal) y realizar la comparación. Por ejemplo, el comparador indica que la entrada analógica es menor que la salida del amplificador, significará que el valor es inferior a 128 con lo cual ya se sabe que se está en la mitad inferior del byte (8 bits) (0 a 127) y por lo tanto el bit 8 debe ser configurado a "0" Lógico, esto ya evita tener que comprar la mitad superior (128 a 255).

El paso siguiente será configurar a 1 el bit 7 y de esta forma verificar si el valor está entre 0 y 63 o entre 64 y 127. Dependiendo de la salida del comparador, se dejará en 1 o en 0 el bit 7. Y así sucesivamente se verificarán los restantes bits (1 a 6). El resultado será el valor digital correspondiente a la conversión.

Este método solo requiere de 8 comparaciones (trabajando con resoluciones de 8 bits) para detectar cualquier valor de entrada a diferencia del método secuencial que requería de 256 comparaciones en el caso extremo.

2 Márgenes dinámicos de entrada

Para conseguir una mejor resolución en los sistemas de medida, se ajusta el rango de la entrada que se pretende adquirir al rango del instrumento. Los rangos de la señal de entrada se refieren a los niveles mínimos y máximos de tensión de entrada que el convertidor puede cuantificar. La mayoría de las tarjetas ofrecen la posibilidad de seleccionar diferentes ganancias y así poder configurar diferentes niveles de rango de voltaje de entrada, se puede configurar para adquirir señales unipolares (señales positivas incluido el cero) y señales bipolares (señales positivas, señales negativas y cero). A continuación (figura 2.12) se muestra una idea de la configuración y ganancia de señales.

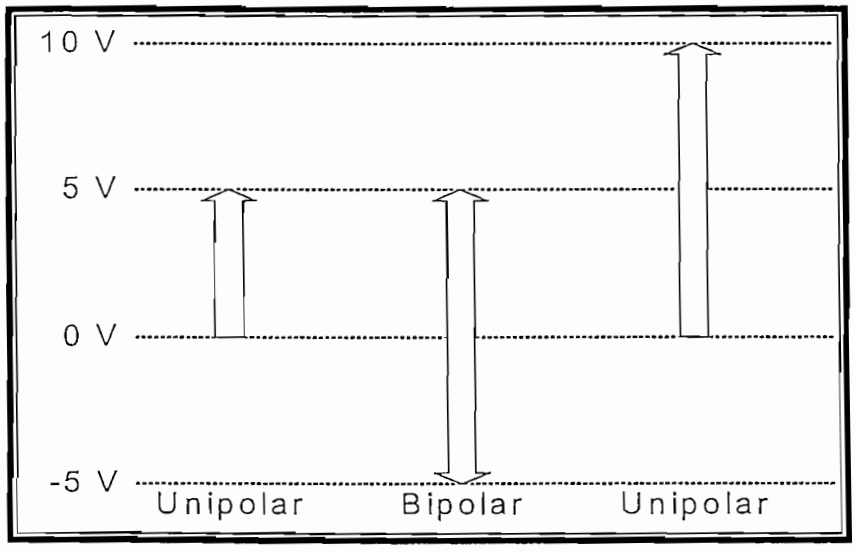


FIGURA 2.12) Señales Unipolares y Bipolares de diferente amplitud

El rango dinámico de la entrada, la resolución y la ganancia disponible determinan la variación más pequeña detectable de señal de entrada.

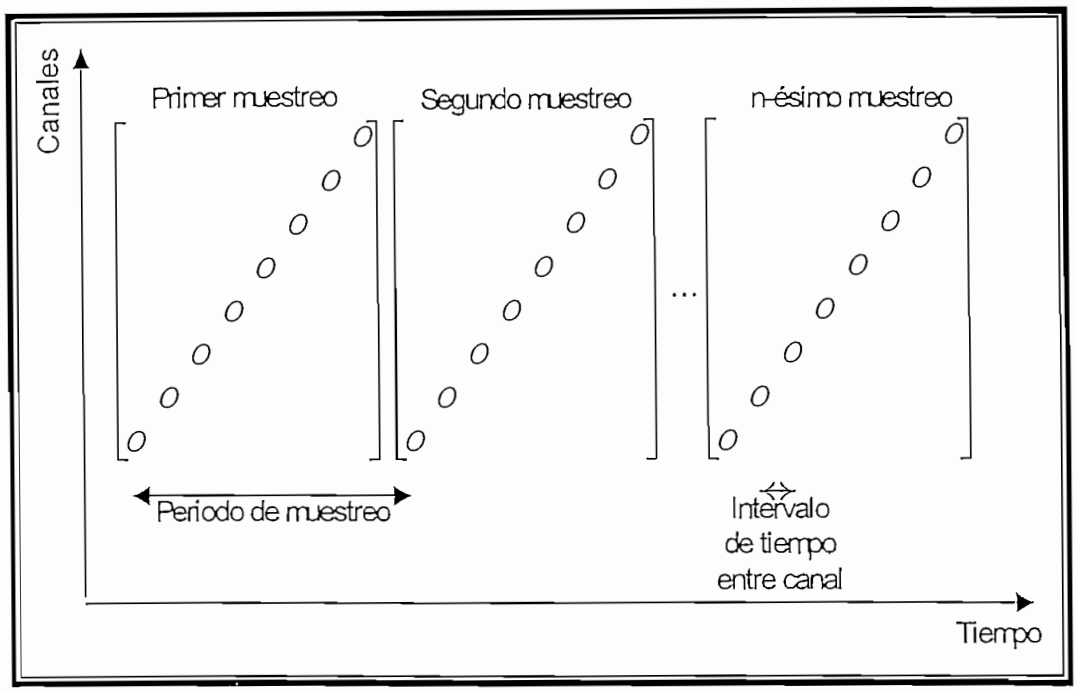


FIGURA 2.14) Muestreo continuo

Otro método es el de muestreo simultáneo, este método consiste en que todos los canales de entrada son muestreados al mismo tiempo (con una diferencia de nanosegundos) ya que cada canal tiene su propia circuiteria de muestreo. Este método es trascendental cuando las relaciones de tiempo de cada señal con las otras es importante (figura 2.15).

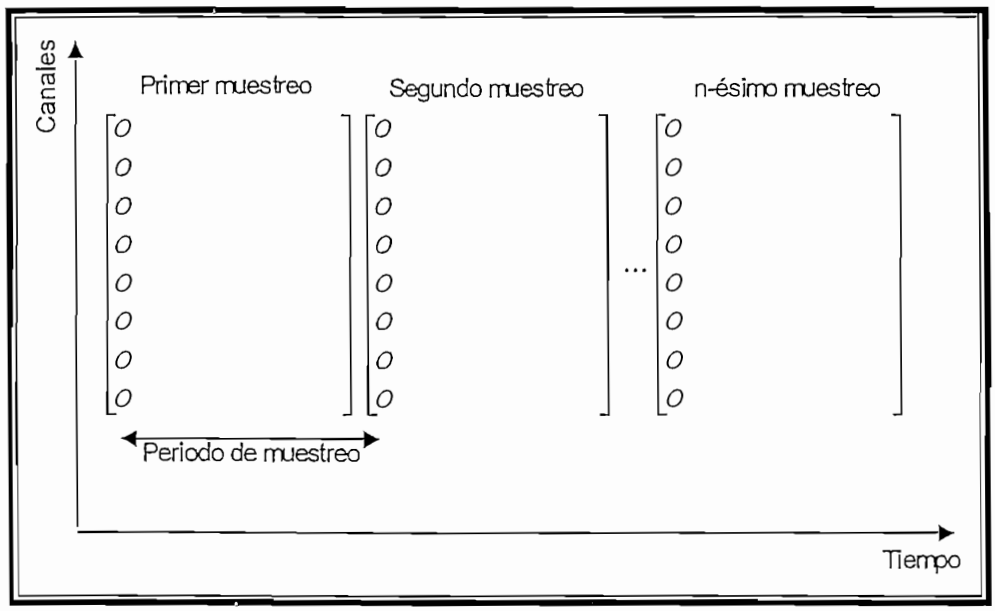


FIGURA 2.15) Muestreo simultáneo

2.6.1.1.2 Salidas analógicas

Estas salidas se utilizan para proporcionar señales de prueba al sistema de adquisición. Uno de los elementos más importantes de esta circuitería es el conversor digital-analógico que determina la calidad de la señal analógica de salida.

Los parámetros que miden esta calidad son el tiempo de asentamiento de la señal, el slew rate y la resolución. El tiempo de asentamiento es el tiempo que necesita la salida para llegar al grado de precisión deseado. El slew rate es el valor máximo de variación de señal que el conversor puede generar a la salida. La resolución a la salida es similar al concepto que ya se ha introducido de resolución a la entrada.

El tiempo de asentamiento y el slew rate determinan con que velocidad puede variar el nivel de la salida del conversor digital-analógico.

1 Conversión Digital - Analógico

Consiste en producir una señal de salida analógica mediante un conjunto de valores digitales, que se envían directamente a un convertidor D/A (Figura 2.16).

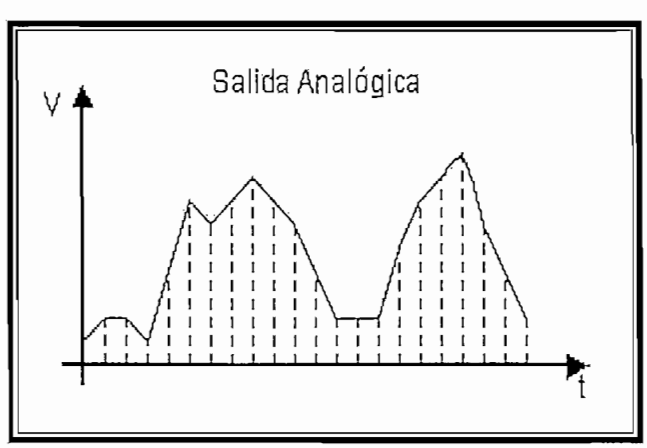


FIGURA 2.16) Señal analógica obtenida del proceso de conversión digital-analógico

2.6.1.1.3 Entradas y salidas digitales

Se utilizan para controlar procesos, generar patrones de prueba y posibilitan la comunicación con los periféricos. Los parámetros más relevantes de esta especificación son el número de líneas digitales, y la velocidad con que los datos pueden entrar y salir.

Entradas digitales: Consiste en leer el Registro de entrada digital (DI) para obtener el valor binario de cada una de las líneas de entrada.

Salidas digitales: Consiste en escribir el Registro de salida digital (DO) para enviar 1s y 0s por cada una de las salidas (líneas) digitales.

Las entradas / salidas digitales (DIO) pueden ser configuradas de dos formas: como entradas (IN) o como salidas (OUT), cuando se configuran como IN éstas solo pueden leer datos, mientras que cuando se configuran como OUT éstas pueden leer y escribir datos.

2.6.1.1.4 Los circuitos de temporización y contadores de entrada y salida

Esta circuitería es útil para contar eventos, medidas temporales de pulsos digitales y la generación de señales cuadradas y de pulsos.

Este tipo de circuitos son necesarios para adquirir las señales en el momento preciso. El disparo (trigger) se utiliza para iniciar y parar la adquisición en función de acontecimientos externos y para sincronizar un proceso de adquisición con otros posibles. La señal de disparo se puede obtener de diferentes fuentes, ya sean internas generadas por las funciones del instrumento que se utiliza, como también pueden ser externas.

2.6.1.2 EJEMPLO DE TARJETA DE ADQUISICIÓN DE DATOS

Como ejemplo se utilizará la tarjeta de adquisición de datos PCI-DAS1000 (Figura 2.17) de Measurement Computing, la misma que es la *daq* utilizada en este proyecto.

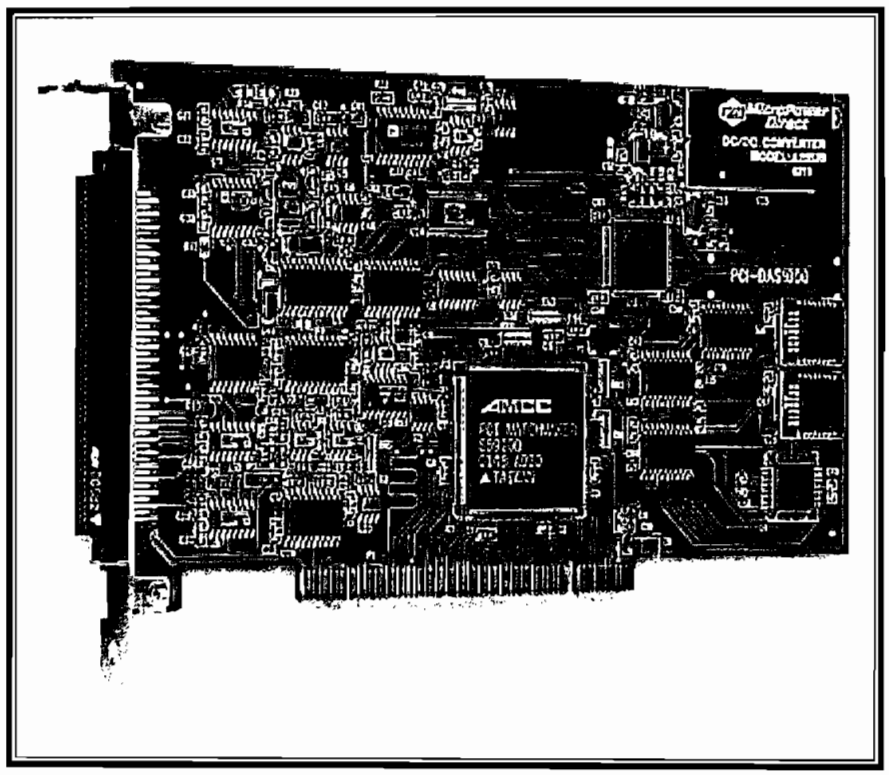


FIGURA 2.17) Tarjeta de adquisición de datos PCI-DAS1000

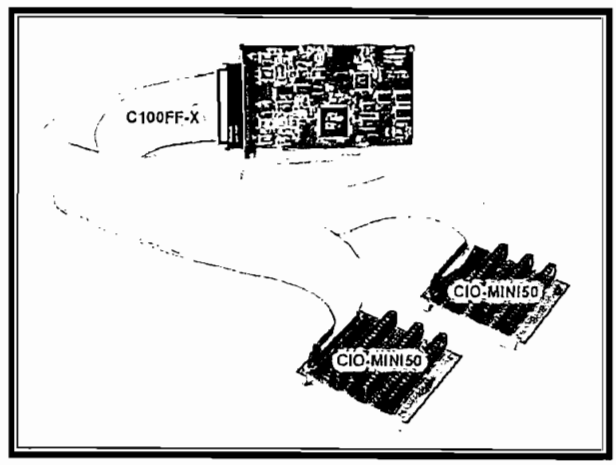


FIGURA 2.18) Cable para interconexión entre Tarjeta PCI y el conector terminal

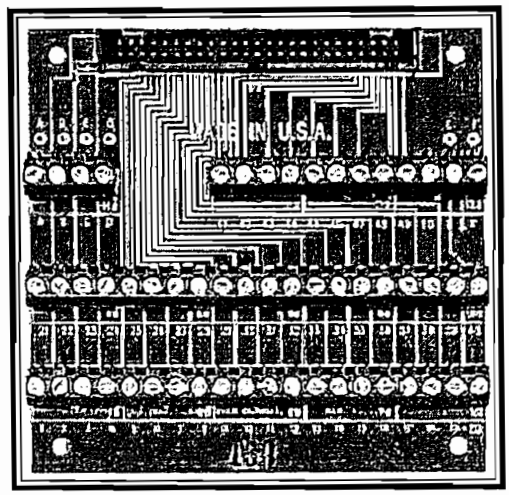


FIGURA 2.19) Conector Terminal de 50 pines

2.6.1.2.1 Características Generales

Consumo de Potencia	
Voltaje: +5 V	Corriente: Típica 0.8 A, Máxima 1 A
Entrada Analógica	
Tipo de conversor A/D	ADS 7800
Resolución	12 bits
Número de Canales	8 diferenciales o 16 simples
Rangos de Entrada (programable)	±10 V, ±5 V, ±2.5 V, ±1.25 V, 0 a 10 V, 0 a 5 V, 0 a 2.5 V, 0 a 1.25 V
Polaridad	Unipolar o Bipolar
Velocidad Máxima de Muestreo	250 KHz
Calibración	Automática, los datos se almacenan en una memoria RAM no volátil que se encuentra en la tarjeta.
Impedancia de Entrada	10 MΩ
Tiempo de conversión A/D	3 μs
Máximo voltaje de entrada	Canales de 1 al 15: -40V a + 55V Canal 0: ±15V
Buffer	1024 palabras de tipo FIFO

Contadores	
Tipo de contador	82C54
Configuración	2 dispositivos 82C54, 3 contadores descendentes por cada 82C54, cada contador de 16 bits
Voltaje de Salida	1L: 3.0 V mínimo 0L: 0.4 V máximo
Voltaje de Entrada	1L: 2.0 V mínimo 0L: 0.8 V máximo
Frecuencia de Entrada	Máximo 10 MHz
Entrada/Salida Digital	
Tipo de Interfaz	82C55A
Configuración	2 grupos de 8 bits y 2 grupos de 4 bits, programable por grupos como entradas o salidas
Número de Canales	24 Entradas/Salidas
Voltaje de Salida	1L: 3.0 V mínimo 0L: 0.4 V máximo
Voltaje de Entrada	1L: 2.0 V mínimo, 5.5 V máximo 0L: 0.8 V máximo, -0.5 V mínomo

TABLA 2.2) Características generales de la PCI-DAS 1000

2.6.1.2.2 Configuración

Mediante software se realiza la debida configuración de la tarjeta; es decir, no necesita de ningún interruptor para definir los tipos de rangos de valores de las entradas analógicas, dirección de las entradas/salidas digitales.

Además se cuenta con el software **Instacal** (Fabricante: Measurement Computing, conocido también como Computer Boards), el cual permite calibrar el

conversor analógico – digital, realizar pruebas de adquisición de valores analógicos y agregar nuevas tarjetas de adquisición.

2.6.1.2.3 Diagrama de bloques

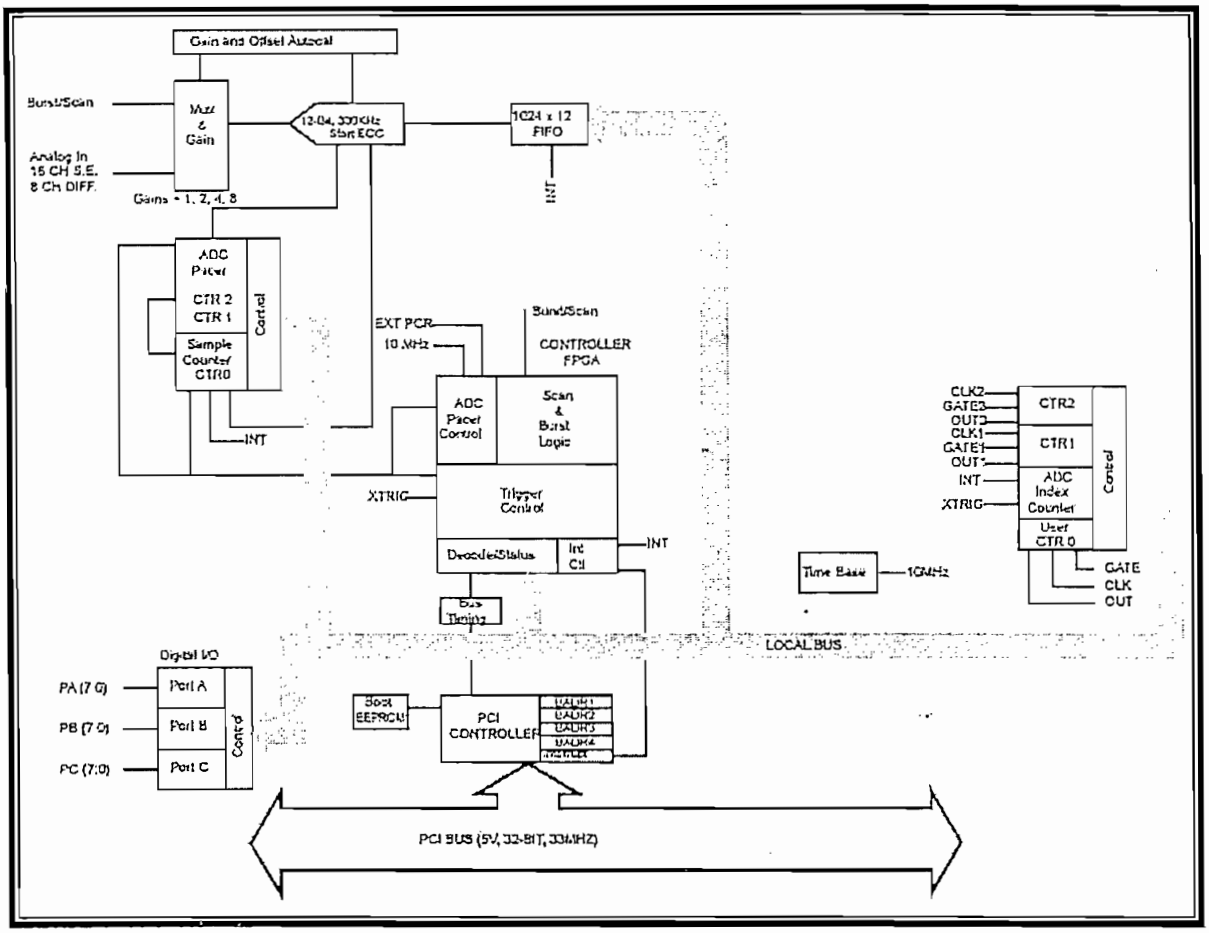


FIGURA 2.20) Diagrama de bloques interno de la PCI-DAS1000

2.6.1.2.4 Interfaz 82C55

Es un interfaz paralelo de 24 líneas, las cuales pueden ser programadas como entradas, salidas o entradas/salidas. Las líneas son compatibles con los niveles TTL (5 V), este interfaz requiere de 4 bytes de direcciones. Puede trabajar en tres modos de operación:

- Modo 0, las 24 líneas son divididas en dos grupos de ocho líneas y dos grupos de 4 líneas, cada grupo puede ser configurado mediante software como todas entradas o todas salidas.

- Modo 1, las 24 líneas son divididas en dos grupos de ocho líneas, mientras que las ocho restantes realizan las funciones de control de los dos grupos antes mencionados. La dirección de los datos es configurada mediante el registro de control.

- Modo 2 crea un bus bidireccional en ocho de las líneas, cinco líneas adicionales proveen el estatus y control de información para el bus.

A continuación se muestra la disposición de pines en el circuito integrado:

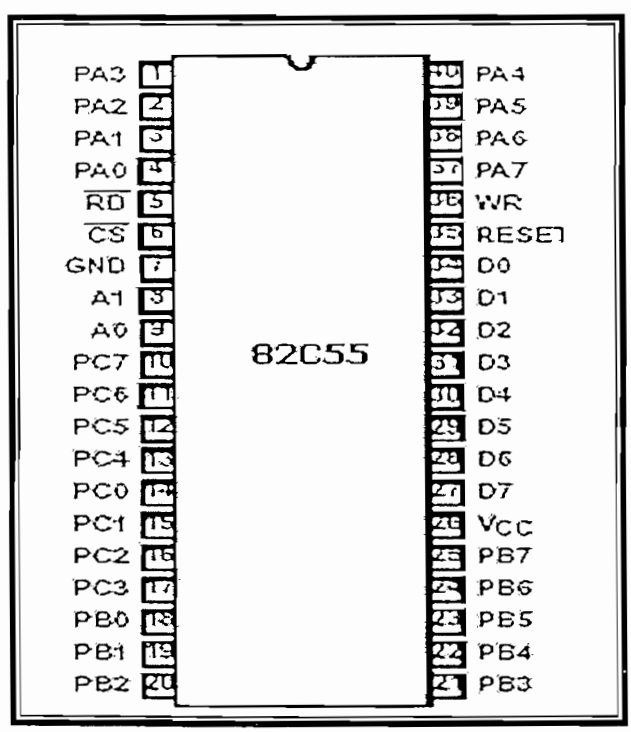


FIGURA 2.21) Diagrama de pines del interfaz 82C55

1 Tabla de Verdad

A_1	A_0	$\overline{RD}$	$\overline{WR}$	$\overline{CS}$	
					Operación: Entrada (Lectura)
0	0	0	1	0	Datos Puerto A ⇒ Bus de Datos
0	1	0	1	0	Datos Puerto B ⇒ Bus de Datos
1	0	0	1	0	Datos Puerto C ⇒ Bus de Datos
					Operación: Salida (Escritura)
0	0	1	0	0	Bus de Datos ⇒ Datos Puerto A
0	1	1	0	0	Bus de Datos ⇒ Datos Puerto B
1	0	1	0	0	Bus de Datos ⇒ Datos Puerto C
1	1	1	0	0	Bus de Datos ⇒ Datos a Control
					Operación: Desabilitación
X	X	X	X	1	Bus de Datos ⇒ 3 estados
1	1	0	1	0	Condición Ilegal
X	X	1	1	0	Bus de Datos ⇒ 3 estados

TABLA 2.3) Tabla de verdad del 82C55

2 Diagrama de Bloques

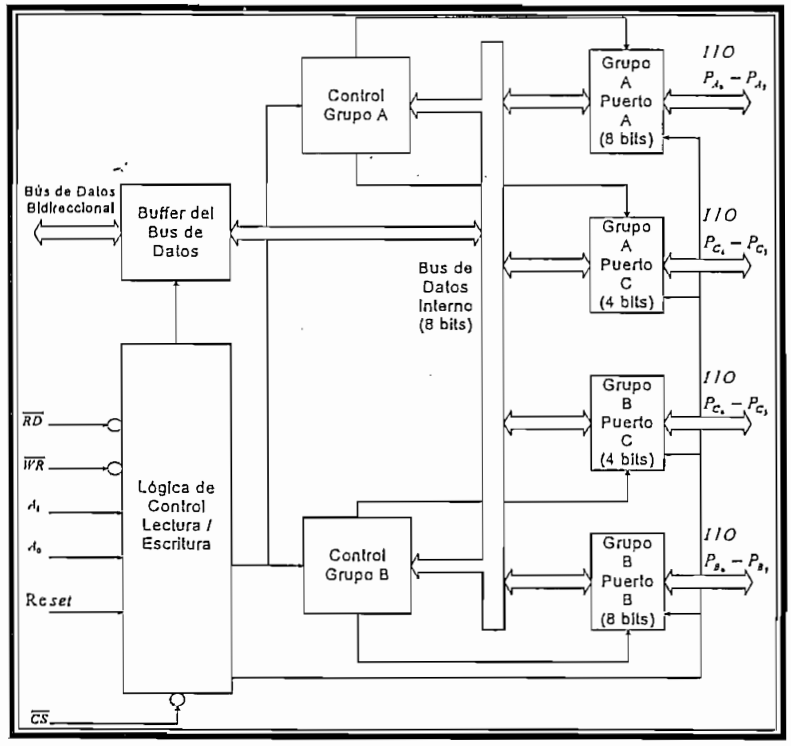


FIGURA 2.22) Diagrama de bloques interno del interfaz 82C55

2.6.1.2.5 Interfaz 82C54

Este interfaz está conformado por tres contadores de 16 bits independientes, éstos pueden trabajar a frecuencias de oscilación de 8, 10 o 12 MHz.

La distribución de pines en el circuito integrado se muestra a continuación:

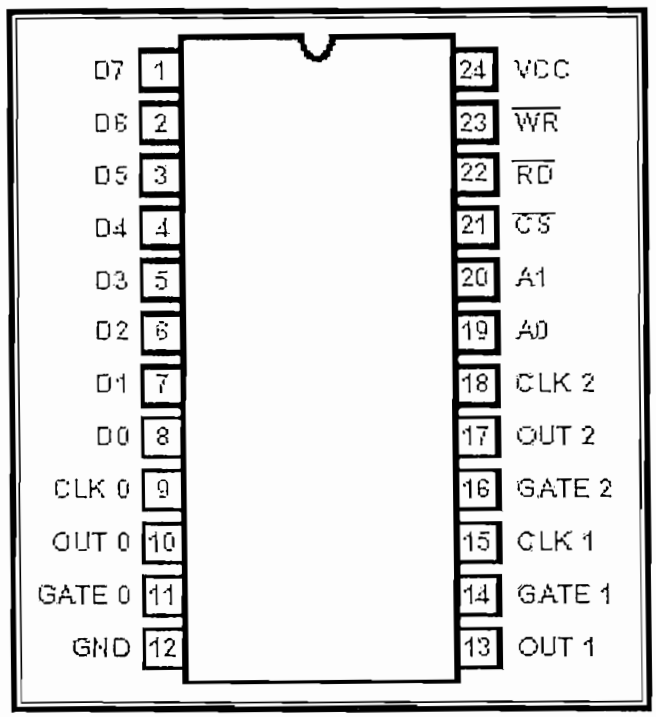


FIGURA 2.23) Diagrama de pines del interfaz 82C54

Símbolo	# de PIN	Tipo	Definición
D7– D0	1 – 8	I/O	DATA: líneas bi-direcionales de bus de tres estados, conectados al sistema de Bus Local.
CLK 0	9	I	Reloj de entrada del contador 0.
OUT 0	10	O	Salida del contador 0
GATE 0	11	I	Compuerta de entrada del contador 0

<table>
<tr><td>GND</td><td>12</td><td></td><td>Tierra</td></tr>
<tr><td>OUT 1</td><td>13</td><td>O</td><td>Salida del contador 1</td></tr>
<tr><td>GATE 1</td><td>14</td><td>I</td><td>Compuerta de entrada del contador 1</td></tr>
<tr><td>CLK 1</td><td>15</td><td>I</td><td>Reloj de entrada del contador 1</td></tr>
<tr><td>GATE 2</td><td>16</td><td>I</td><td>Compuerta de entrada del contador 2</td></tr>
<tr><td>OUT 2</td><td>17</td><td>O</td><td>Salida del contador 2</td></tr>
<tr><td>CLK 2</td><td>18</td><td>I</td><td>Reloj de entrada del contador 2</td></tr>
<tr><td>A0,A1</td><td>19-20</td><td>I</td><td>ADDRESS: Selecciona con cual de los tres contadores se va a trabajar o las operaciones de read/write en el Control Word Register.
<table>
<tr><td>A1</td><td>A0</td><td>Selección</td></tr>
<tr><td>0</td><td>0</td><td>Contador 0</td></tr>
<tr><td>0</td><td>1</td><td>Contador 1</td></tr>
<tr><td>1</td><td>0</td><td>Contador 2</td></tr>
<tr><td>1</td><td>1</td><td>Control Word Register</td></tr>
</table></td></tr>
<tr><td>CS</td><td>21</td><td>I</td><td>CHIP SELECT: habilitación o deshabilitación del 82C54</td></tr>
<tr><td>RD</td><td>22</td><td>I</td><td>READ: durante operaciones de lectura</td></tr>
<tr><td>WR</td><td>23</td><td>I</td><td>WRITE: Durante operaciones de escritura</td></tr>
<tr><td>Vcc</td><td>24</td><td></td><td>Voltaje de alimentación</td></tr>
</table>

TABLA 2.4) Tabla de verdad del 82C54

2.6.1.2.6 Conversor Analógico – Digital ADS 7800

El ADS 7800 es un conversor analógico / digital de 12 bits, la conversión la realiza mediante el método de aproximaciones sucesivas, la velocidad de muestreo de es 333 KHz, los rangos de entrada pueden ser de ± 5V o ± 10V, la potencia máxima de disipación es de 215 mW

A continuación se muestra la disposición de pines en el circuito integrado:

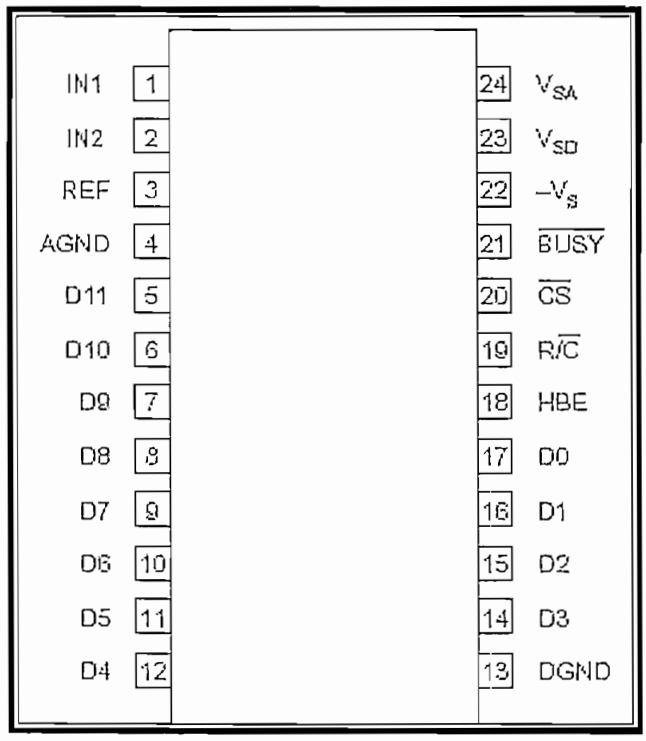


FIGURA 2.24) Diagrama de pines del ADC ADS 7800

1 *Diagrama de Bloques*

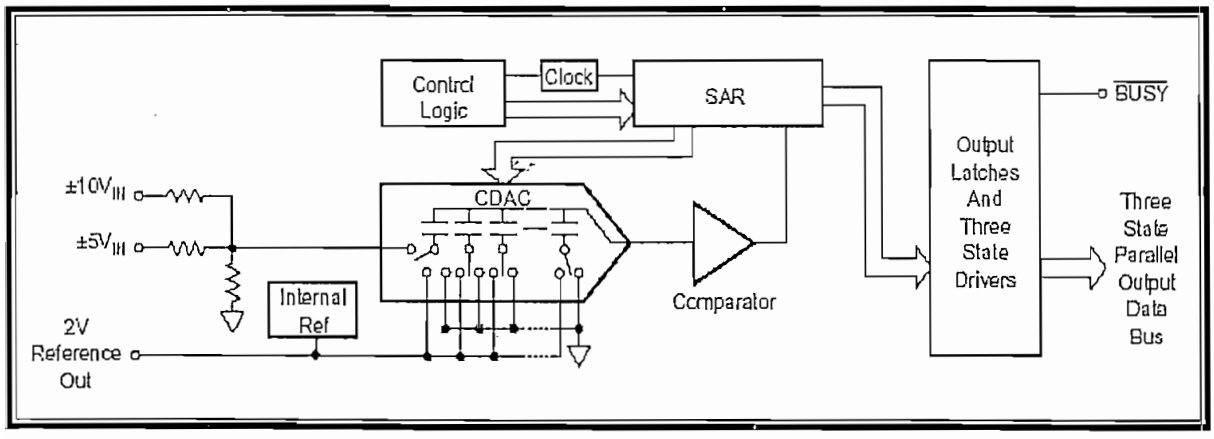


FIGURA 2.25) Diagrama de bloques interno del ADS 7800

2.6.2 POTENCIÓMETROS

Un potenciómetro es un resistor sobre el cual desliza un contacto eléctrico en un punto intermedio de su elemento resistivo, el cual se lo utiliza como divisor de voltaje entre los terminales fijos.

2.6.2.1 Símbolo:

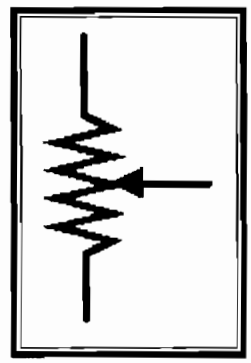


FIGURA 2.26) Símbolo que representa a un potenciómetro

En la figura 2.27 se muestra diferentes tipos de potenciómetros disponibles en el mercado:

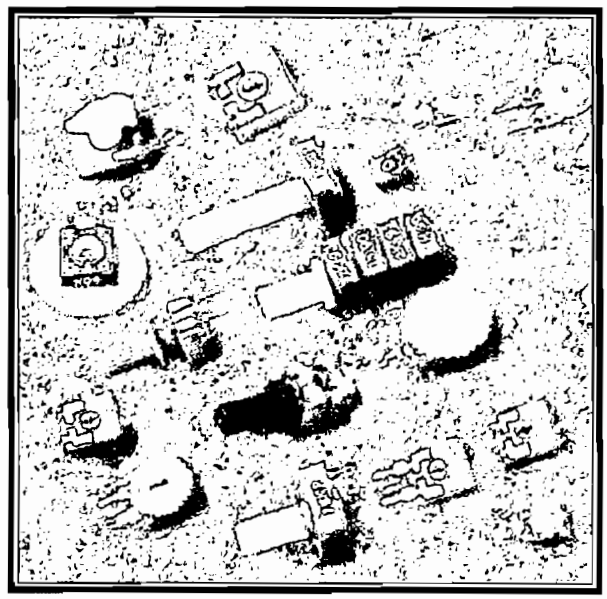


FIGURA 2.27) Diversos tipos de potenciómetros

2.6.2.2 Potenciómetros Digitales

Son circuitos integrados en los que mediante impulsos eléctricos a un pin conectan un terminal (cursor) a un punto intermedio de una cadena de resistores.

Como se mencionó en el numeral 2.6 (Elementos utilizados) los potenciómetros empleados para el desarrollo de este proyecto son: X9C503, que se describen a continuación:

2.6.2.2.1 Características Generales

- ✓ Puede tener 100 valores diferentes de resistencia
- ✓ Bajo consumo de potencia
- ✓ Tiempo de retención de datos del registro (memoria) de 100 años

Este potenciómetro consta de un arreglo de resistencias, un conjunto de transistores cumpliendo la función de interruptores, una sección de control y una memoria no-volátil. El potenciómetro está conformado por 99 resistencias[9] y una red de conmutación, la posición del terminal variable R_W es controlado por los terminales $\overline{CS}$, $U/\overline{D}$ y $\overline{INC}$, la posición en la que se encuentra el terminal R_W es siempre almacenada en la memoria no-volátil. Cada vez que el potenciómetro es energizado el terminal R_W toma el valor que se encontraba almacenado en la memoria no-volátil.

Este integrado puede ser usado como potenciómetro (tres terminales) o como resistencia variable (dos terminales), en la figura 2.28 se puede observar este potenciómetro.

---

[9] Para el caso del X9C503 cada resistencia tiene un valor de 500 Ω

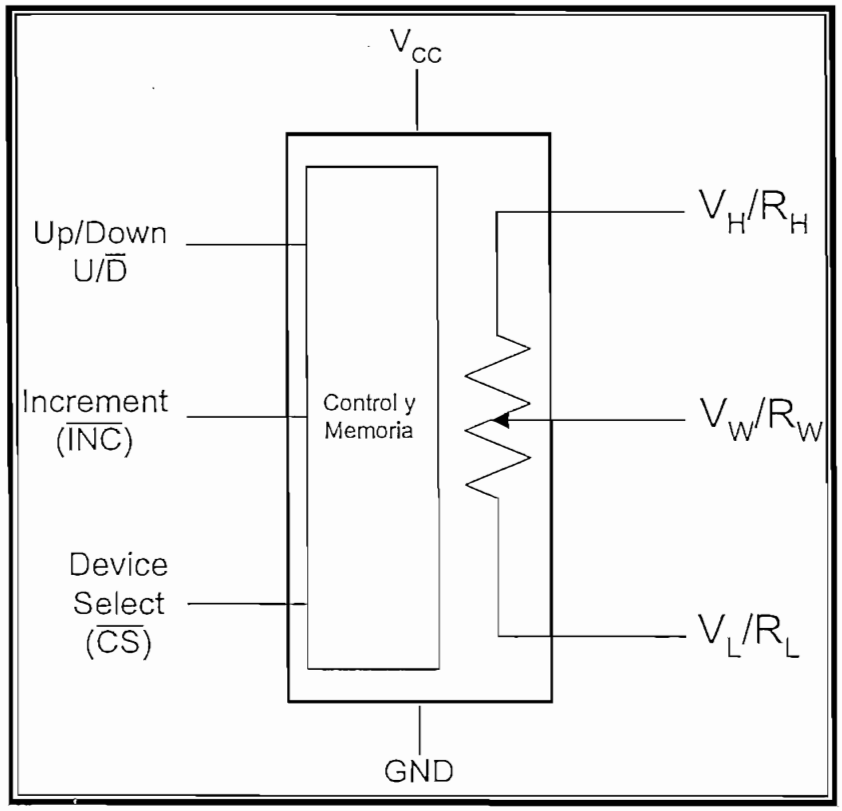


FIGURA 2.28) Diagrama general de un potenciómetro digital Xicor X9Cxxx

En la figura 2.29 se muestra el diagrama de bloques del potenciómetro digital X9Cxxx; es decir, las partes que componen mencionado integrado:

- ✓ Contador Up/Down de 7 bits
- ✓ Memoria no-volátil de 7 bits
- ✓ Circuito de control
- ✓ Decoder
- ✓ Arreglo de transistores
- ✓ Arreglo de resistencias

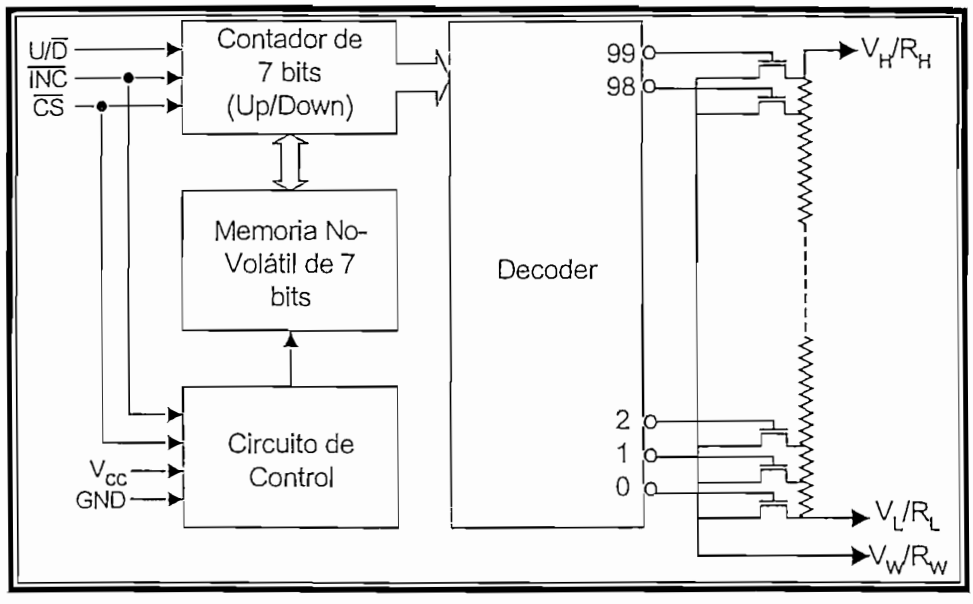


FIGURA 2.29) Diagrama de bloques interno del potenciómetro digital Xicor X9Cxxx

Pin	Símbolo	Descripción
1	INC	Incremento: En la transición del 1L a 0L el valor de resistencia se incrementa o decrementa dependiendo del valor del terminal U/D.
2	U/D	Up/Down: Entrada de control, la cual dirige el movimiento del terminal variable, es decir si se incrementa o decrementa el valor del resistencia del potenciómetro.
3	R_H/V_H	Terminal Alto: es equivalente al terminal fijo de un potenciómetro mecánico. Se lo considera alto con respecto a la dirección de movimiento seleccionada en la entrada U/D.
4	Vss	Tierra
5	V_W/R_W	Terminal Variable: es equivalente al terminal variable de un potenciómetro mecánico. La posición en la que se encuentra este terminal es determinado por las entradas de control.
6	R_L/V_L	Terminal Bajo: es equivalente al terminal fijo de un potenciómetro mecánico. Se lo considera bajo con respecto a la dirección de movimiento seleccionada en la entrada U/D.
7	CS	Chip Select: Habilitación del Integrado, esta entrada necesita estar en bajo (0L) para habilitar el funcionamiento del integrado.
8	Vcc	Alimentación de Voltaje de polarización.

TABLA 2.5) Tabla de función de cada uno de los terminales del X9C503

2.7 MONTAJE, PRUEBAS Y AJUSTES

El montaje de la tarjeta se lo realiza de la siguiente forma:

1. Quitar la carcaza del computador;

2. En un slot PCI del computador que esté disponible colocar la tarjeta[10];

3. Tapar el computador;

4. Encender el computador; al realizar este proceso se reconoce automáticamente que un nuevo dispositivo de hardware ha sido conectado;

5. Luego se debe insertar en el CD-ROM el CD que contiene el controlador de la Tarjeta de Adquisición de Datos.

Después de haber realizado el montaje y la respectiva configuración en el sistema operativo se debe proceder a configurar y calibrar la tarjeta[11].

La prueba a realizarse consiste en medir los valores de voltaje entregados por la tarjeta, ya sea cuando se encuentre en 0_L o 1_L y determinar si se encuentra dentro de los rangos aceptados para diferenciar si se tiene un 0_L o 1_L. Los valores de voltaje entregado por las salidas digitales de interés para este proyecto son:

---

[10] El computador debe estar apagado.

[11] Para la configuración de la tarjeta y calibración del conversor analógico / digital referirse al capítulo 3, en el cual se indica el funcionamiento del software INSTACAL.

Salida Digital			
#	Puerto	0_L [V]	1_L [V]
1	A0	0.001	5.120
2	A1	0.001	5.120
3	A2	0.003	5.120
4	A3	0.001	5.120
5	A4	0.001	5.120
6	A5	0.001	5.120
7	A6	0.001	5.120
8	A7	0.001	5.120

TABLA 2.6) Valores medidos de la PCI-DAS 1000

En este capítulo no se considera realizar las pruebas de las entradas analógicas debido a que son datos que ingresan al computador y serán tratados en el capítulo 3.

2.8 FUNCIONAMIENTO

La tarjeta de adquisición de datos toma y/o envía datos por el conector de 100 pines y se conecta al computador por el bus PCI, la tarjeta se encuentra disponible todo el tiempo, la cual mediante software se puede configurar adecuadamente para luego iniciar la adquisición y/o envío de datos.

CAPÍTULO 3:
INTERFAZ SOFTWARE

3.1 INTRODUCCIÓN

La interfaz de usuario (IU) es uno de los componentes más importantes de cualquier sistema computacional, pues funciona como el vínculo entre el ser humano y la máquina. La interfaz de usuario es responsable de solicitar comandos al usuario, y de desplegar los resultados de la aplicación de una manera comprensible, sin embargo la interfaz de usuario no es responsable de los cálculos de la aplicación, ni del almacenamiento, recuperación y transmisión de la información.

El éxito de un programa frecuentemente se debe al corto tiempo en el que puede aprender el usuario a emplear el software, igualmente importante es lograr que el usuario alcance sus objetivos con el programa de la manera más sencilla posible.

Los programas son usados por usuarios con distintos niveles de conocimientos, desde principiantes hasta expertos, es por ello que no existe una interfaz válida para todos los usuarios y todas las tareas. Debe permitirse libertad al usuario para que elija el modo de interacción que más se adecue a sus objetivos en cada momento. La mayoría de los programas ofrecen varias formas de interacción al usuario.

Existen tres puntos de vista distintos en una IU: el del usuario, el del programador y el del diseñador. Cada uno tiene un modelo mental propio de la interfaz, que contiene los conceptos y expectativas acerca de la misma, desarrollados a través de su experiencia.

Es difícil saber el grado de conocimientos de cómputo del usuario final, lo cual, frecuentemente, hace que las interfaces de usuario desarrolladas no sean las apropiadas. Se da el caso de que el diseñador implemente la interfaz de usuario pensando en que la van a usar programadores avanzados, como el propio

diseñador, razón por la cual, cuando el producto final es usado por el usuario es posible que se presenten una gran cantidad de problemas en su aplicación.

La interfaz de usuario de un programa es un software de una computadora que presenta información al usuario y le permite interactuar con la información y con el computador. También se puede considerar parte de la interfaz de usuario la documentación (manuales, ayuda, tutoriales) que acompaña al software.

Si la interfaz de usuario está bien diseñada, el usuario encontrará la respuesta que espera a su acción. Si no es así puede ser frustrante su operación, ya que el usuario habitualmente tiende a culparse a sí mismo por no saber usar el software.

3.2 DEFINICIÓN

La interfaz del usuario permite comunicarnos con los sistemas y que los sistemas nos indiquen los resultados y/o las necesidades que surjan de esa comunicación.

Sin una buena interfaz de usuario, el sistema más óptimo, es prácticamente inútil, ya que al ser difícil de utilizar toda su potencialidad no puede transformarse en una acción concreta en el mundo real.

La interfaz de usuario es un conjunto de protocolos y técnicas para el intercambio de información entre una aplicación computacional y el usuario.

La interfaz software se puede definir también como el conjunto de trabajos y pasos que seguirá el usuario, durante todo el tiempo que se relacione con el programa, detallando lo que verá y escuchará en cada momento, y las acciones que realizará, así como también las respuestas que el sistema le otorgará.

3.3 DESCRIPCIÓN

El software utilizado para el desarrollo del interfaz es MATLAB (Matrix Laboratory) versión 6.0, MatLab es un lenguaje técnico computacional de alta calidad que permite, ente otras cosas, adquirir y enviár datos por medio de la tarjeta de adquisición de datos, mostrar gráficamente los datos adquiridos por la *daq*, además permite crear interfaces de usuario para permitir manipular datos.

3.3.1 HERRAMIENTA DE ADQUISICIÓN DE DATOS

MatLab cuenta con una herramienta para la adquisición de datos denominada *Data Acquisition Toolbox*, con la cual es posible la comunicación entre el software MatLab y el software controlador de la tarjeta de Adquisición de datos.

La herramienta de adquisición de datos soporta tarjetas de adquisición de los siguientes fabricantes: National Instruments (tarjetas de la serie E y de la serie 1200), Computer Boards, Agilent Technologies y las tarjetas de sonido de Windows.

Esta herramienta está conformada por tres componentes, los cuales son: funciones (almacenadas en archivos de extensión ".m"), el motor de adquisición de datos y de un adaptador para el manejo del hardware.

A continuación se describe la función de la herramienta de adquisición de datos con la que cuenta MatLab y además la figura 3.1 se muestra gráficamente como fluye la información (valor de las propiedades, datos y eventos) de un elemento a otro:

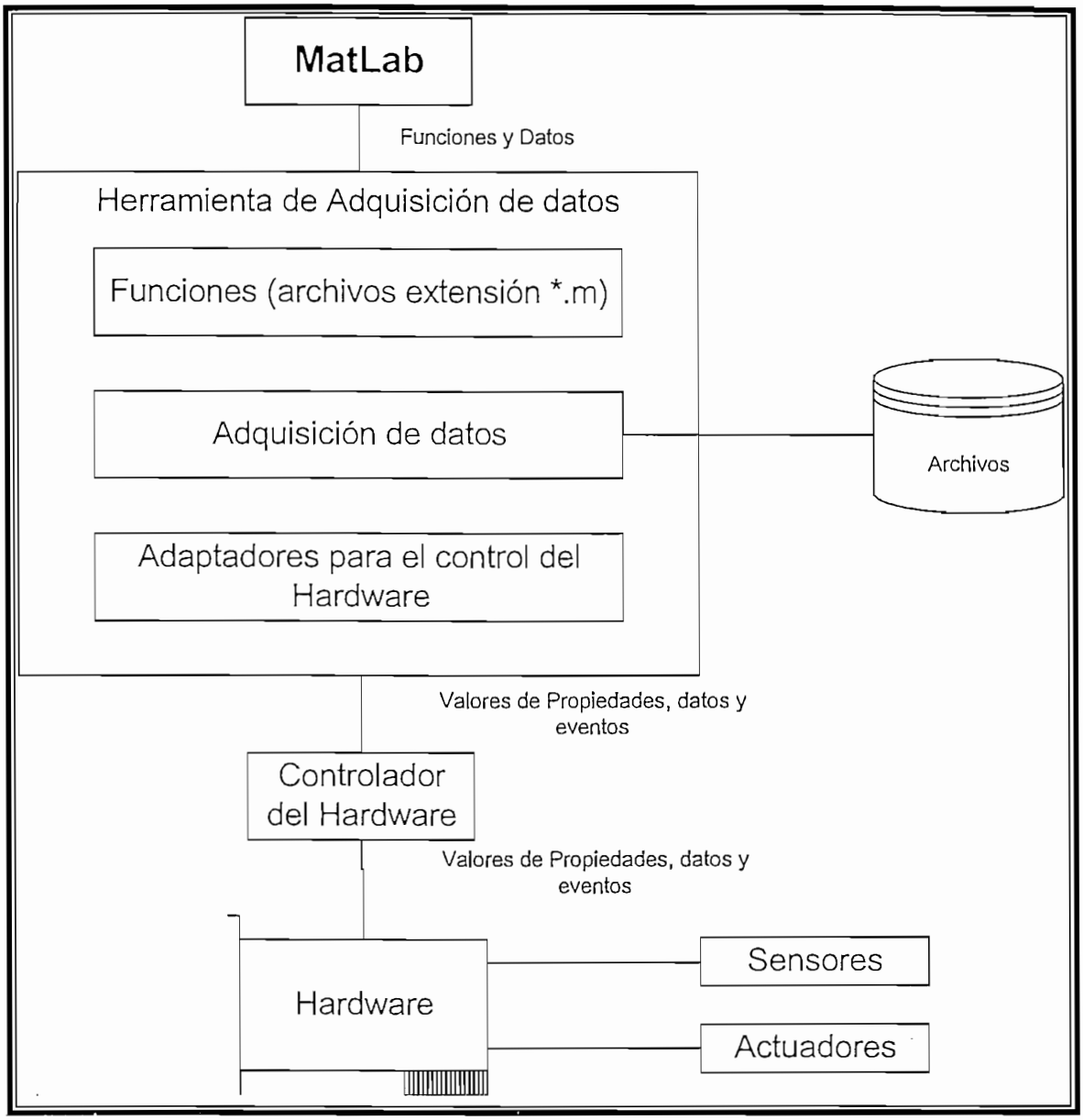


FIGURA 3.1) Proceso del Software en la Adquisición de datos

3.3.1.1 Funciones

Para realizar la adquisición de datos se hace el llamado a funciones, las cuales son almacenadas en archivos de extensión .m, las funciones puedes ser: crear los objetos (entradas analógicas, salidas analógicas, entradas / salidas digitales), adquirir o enviar datos, configurar valores de propiedades.

3.3.1.2 Motor de Adquisición de datos

El motor de Adquisición de datos es un archivo de extensión .dll (dynamic link library), en este archivo se almacenan los objetos y los valores de las propiedades asociadas a estos objetos que se controlan con la aplicación creada para la adquisición de datos, controla la sincronización de los eventos y controla el almacenamiento de los datos, mientras el motor realiza estas tareas se puede usar el programa para analizar los datos adquiridos.

3.3.1.3 Adaptador para el manejo de Hardware

Este adaptador es un interfaz entre el motor de adquisición de datos y el controlador del Hardware, la función de este adaptador es intercambiar la información entre MatLab y la tarjeta de adquisición de datos (por medio del controlador de hardware, el cual depende del proveedor de la tarjeta de adquisición de datos, para el caso de este proyecto el nombre del adaptador es CBI[12])

3.3.1.4 Información de la herramienta de Adquisición de datos

Para observar información general de la herramienta de adquisición de datos en MatLab se escribe la siguiente línea:

```
>> daq=daqhwinfo

daq =

          ToolboxName: 'Data Acquisition Toolbox'
       ToolboxVersion: [1x39 char]
        MatlabVersion: '6.0.0.88 (R12)'
    InstalledAdaptors: {2x1 cell}
```

FIGURA 3.2) Comando para obtener información de la herramienta de Adquisición de datos

[12] Para el caso de las tarjetas de sonido de Windows el nombre del adaptador es "winsound"

Donde "daq" es el nombre de la variable donde se almacena la información

Para saber que Adaptadores se tiene instalados en el computador se debe escribir el siguiente comando:

```
>> daq.InstalledAdaptors

ans =

    'cbi'
    'winsound'
```

FIGURA 3.3) Comando para obtener información cuales son las DAQ que se encuentran instaladas

Para tener más información acerca del adaptador para la tarjeta de adquisición de datos o de la tarjeta de sonido se escribe el siguiente comando:

```
>> daq=daqhwinfo('cbi')

daq =

           AdaptorDllName: [1x46 char]
        AdaptorDllVersion: 'Version 2.0   (R12) 05-Oct-2000'
              AdaptorName: 'cbi'
               BoardNames: {'PCI-DAS1000'}
        InstalledBoardIds: {'1'}
    ObjectConstructorName: {1x3 cell}
```

FIGURA 3.4) Comando para obtener información de la DAQ de Computer Boards

```
>> daq=daqhwinfo('winsound')

daq =

           AdaptorDllName: [1x51 char]
        AdaptorDllVersion: 'Version 2.0   (R12) 05-Oct-2000'
              AdaptorName: 'winsound'
               BoardNames: {'Audio integrado de Intel(r)'  [1x18 char]}
        InstalledBoardIds: {'0'  '1'}
    ObjectConstructorName: {2x2 cell}
```

FIGURA 3.5) Comando para obtener información de la Tarjeta de sonido

Para saber cuales son los subsistemas con los que cuenta la tarjeta de adquisición de datos o la tarjeta de sonido se escribe la siguiente línea de comando:

```
>> daq.ObjectConstructorName(:)

ans =

    'analoginput('cbi',1)'
    'digitalio('cbi',1)'
```

FIGURA 3.6) Comando para obtener información de los subsistemas con los que cuenta la DAQ "CBI"

```
>> daq.ObjectConstructorName(:)

ans =

    'analoginput('winsound',0)'
    'analoginput('winsound',1)'
    'analogoutput('winsound',0)'
    'analogoutput('winsound',1)'
```

FIGURA 3.7) Comando para obtener información de los subsistemas con los que cuenta la DAQ "Winsound"

3.3.2 SOFTWARE DE CONFIGURACIÓN Y PRUEBAS DE LA TARJETA DE ADQUISICIÓN DE DATOS

Las tarjetas de adquisición de datos cuentan con programas que permiten manipular propiedades del hardware, como por ejemplo: calibrar las entradas analógicas y salidas analógicas, realizar pruebas referente a la adquisición de datos, además permiten agregar o eliminar tarjetas de adquisición de datos con sus respectivos controladores.

Para el caso de las tarjetas de adquisición de datos de Computer Boards el software desarrollado para cumplir el fin descrito en el párrafo anterior se denomina *Instacal*.

Para el caso de tarjetas de adquisición de datos de otros fabricantes como lo es National Instruments la herramienta (software) desarrollada se denomina *Measurement and Automation Explorer*.

En la siguiente figura se muestra la pantalla principal de la herramienta *Instacal*, esta herramienta es muy amigable para el usuario; es decir, es fácil de comprender su funcionamiento y lo principal es muy sencillo usarlo.

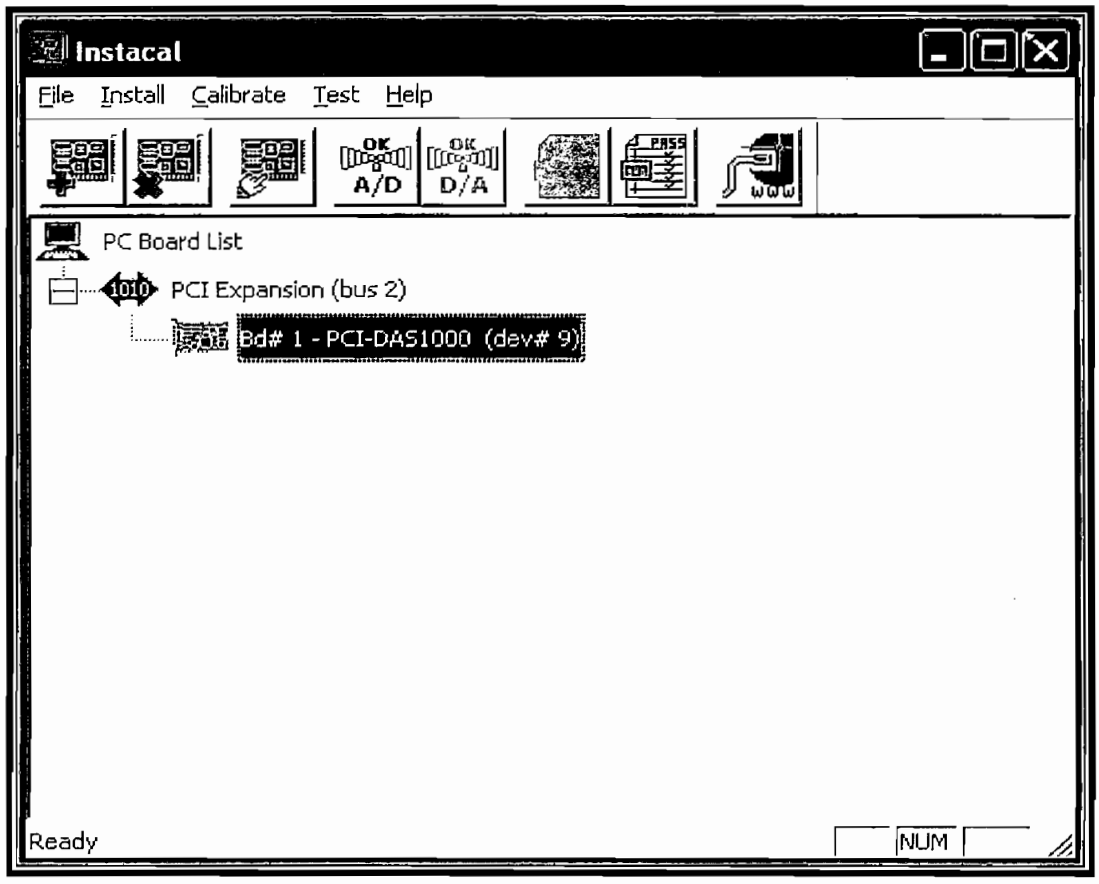


FIGURA 3.8) Software *Instacal* para instalación y configuración de las DAQ de Computer Boards

Si se requiere agregar una tarjeta de adquisición de datos de este fabricante se debe hacer un clic con el ratón en la opción *Add Board*, como se muestra en la siguiente figura.

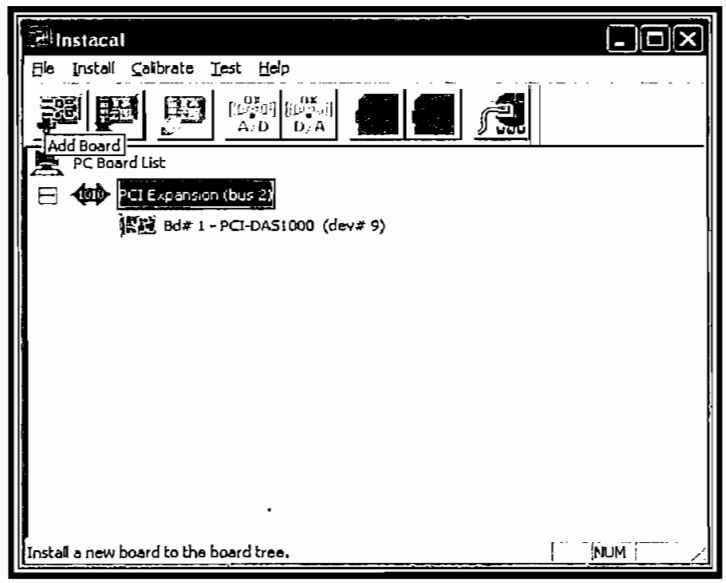


FIGURA 3.9) Agregar una nueva DAQ de Computer Boards

Después de haber realizado la acción descrita anteriormente se procede a seleccionar el modelo de la tarjeta que se instaló en el computador, esta pantalla muestra además los números de tarjeta (posiciones) que se encuentran ocupadas por otras tarjetas y permite seleccionar la posición en la que se desea configurar la nueva tarjeta.

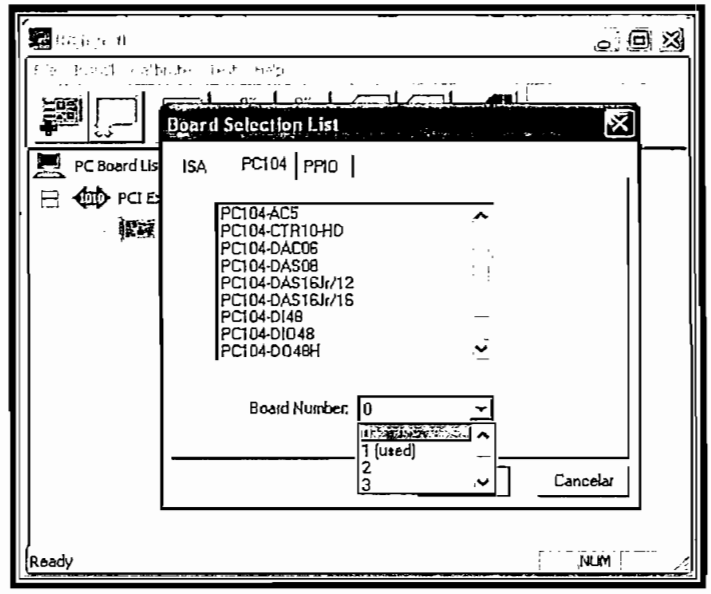


FIGURA 3.10) Seleccionar la DAQ y el número de tarjeta que se va a agregar

Para configurar diferentes propiedades de una tarjeta específica se debe seleccionar la opción *Configure...*

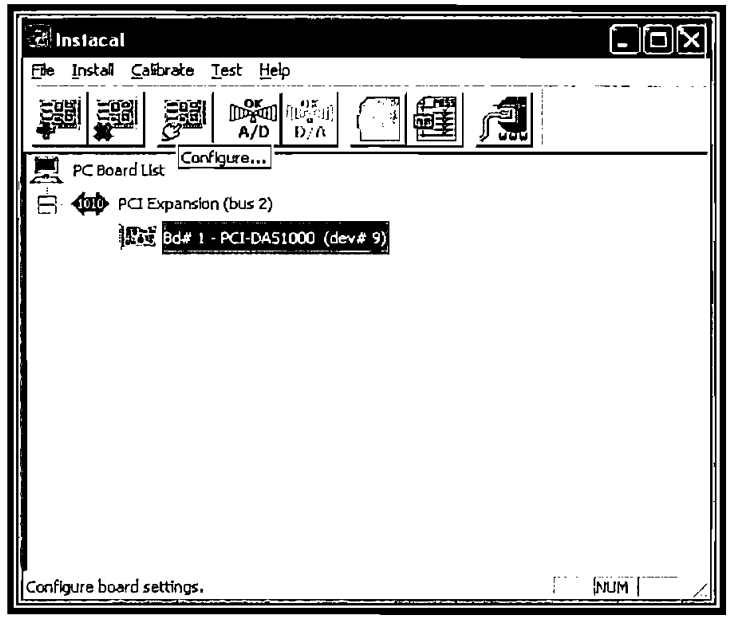


FIGURA 3.11) Configuración de una DAQ de Computer Boards

La opción de configuración permite seleccionar el número de canales con los que se requiere trabajar; es decir, 8 canales diferenciales o 16 canales con referencia a tierra.

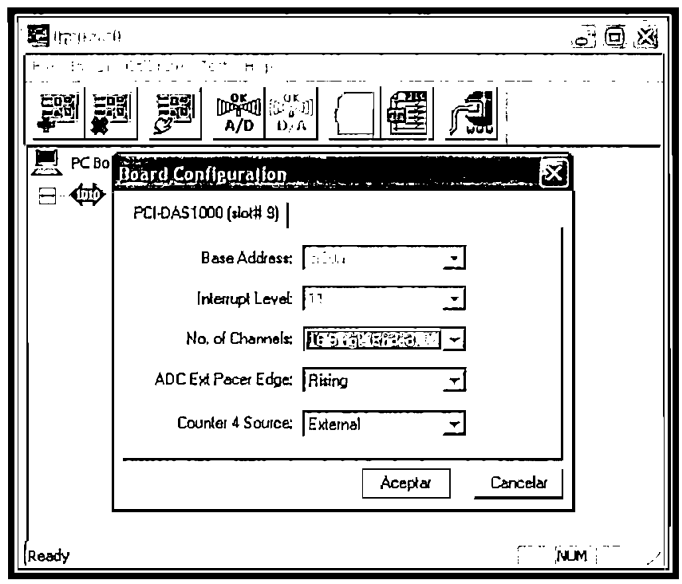


FIGURA 3.12) Configuración de canales, fuente de contador, etc.

Además permite calibrar el conversor analógico – digital, el cual se utiliza para la adquisición de datos, la calibración se la realiza en base a una señal generada por la propia tarjeta. El proceso de calibración del conversor A/D se lo realiza para los diferentes niveles de amplitud, para el caso de la tarjeta PCI-DAS 1000 los niveles son: 10 V , 5 V, 2.5 V y 1.25 V.

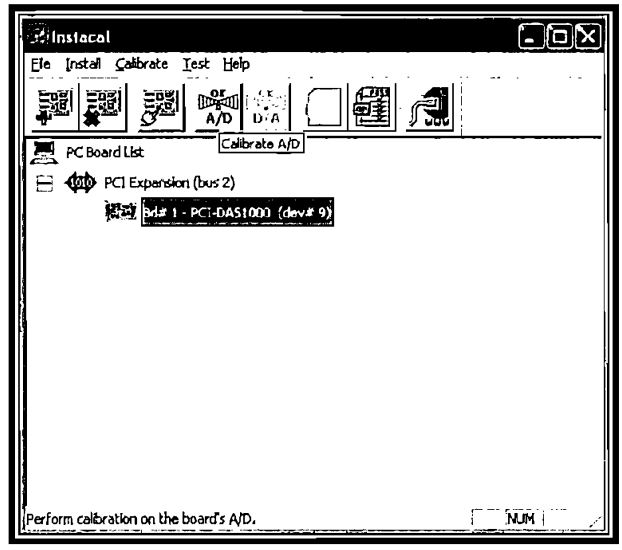


FIGURA 3.13) Calibración de el conversor Analógico / Digital

Para la calibración del conversor analógico – digital, el software automáticamente realiza la calibración.

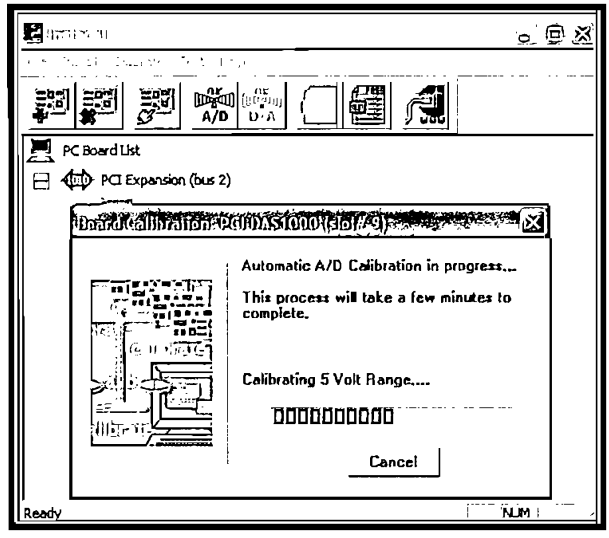


FIGURA 3.14) Ejecución de la calibración del ADC para los diferentes rangos de entrada

Como se mencionó anteriormente, con la herramienta *Instacal* se pueden realizar pruebas de la adquisición de datos, se genera una señal cuadrada con una de las entradas / salidas digitales y se adquiere dicha señal por una de las entradas analógicas, para realizar esta prueba se debe configurar:

- ✓ El canal que se requiere que adquiera la señal,
- ✓ La salida digital, el contador o señal externa que generará la señal y,
- ✓ La amplitud de la señal.

Además se debe realizar las respectivas conexiones físicas como se muestran en la figura.

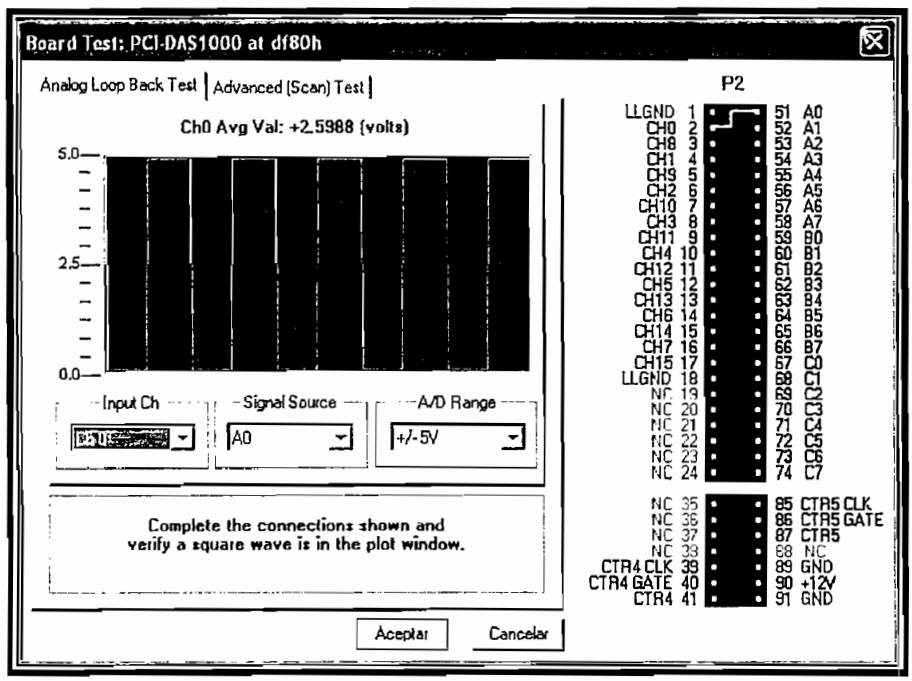


FIGURA 3.15) Pruebas con la DAQ: generación de señales y adquisición de las mismas

En las pruebas también se puede obtener los datos de una o varias señales muestreadas, referirse a la siguiente figura.

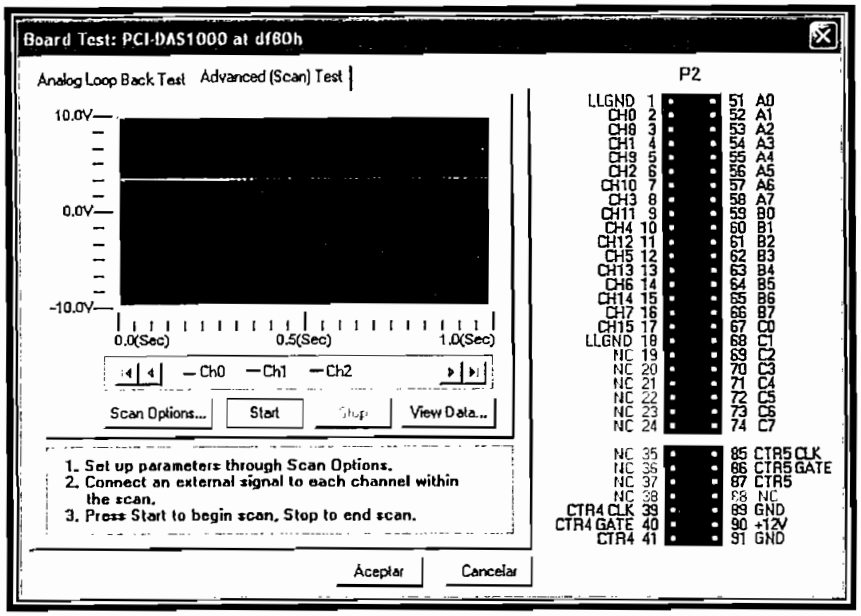


FIGURA 3.16) Pruebas con la DAQ: muestreo de diferentes señales de entradas

3.3.3 COMENZAR A TRABAJAR CON LAS TARJETAS DE ADQUISICIÓN DE DATOS

A continuación se muestra las líneas de programa que se deben escribir para crear los diferentes objetos, los cuales permitirán obtener y enviar datos.

3.3.3.1 Adquisición de Datos

El proceso de adquisición de datos se realiza de la siguiente forma:

1. Creación del Objeto: Con la siguiente instrucción se realiza la creación de la entrada analógica a la cual se la denominará *ai*. Para la adquisición de datos se utilizará la tarjeta de sonido de windows.

$$ai = analoginput('winsound')$$

2. Agregar canales: Después de haber creado el objeto se debe agregar los canales, los cuales serán los caminos por donde se realizará la adquisición de datos, para el caso de la tarjeta de sonido máximo se

puede agregar dos canales, ya que es el número de canales de entrada con los que cuenta. El número de canales a agregar es dos.

$addchannel(ai,1:2)$

3. Configurar valores de las propiedades: Después de haber agregado canales al objeto se debe configurar la velocidad de muestreo (44.1 KHz) y el número de muestras que se toman cada vez que se realiza el muestreo (8000); es decir, se muestrea una señal 44100 veces en un segundo y cada vez que se muestrea la señal se toman 8000 muestras.

$set(ai,'SampleRate',44100)$
$set(ai,'SamplesPerTrigger',8000)$

4. Iniciar de la Adquisición de datos:

$start(ai)$
$getdata(ai)$

5. Borrar: Cuando ya no se necesite el objeto es mejor borrar y limpiar, es decir liberar memoria

$delete(ai)$
$clear('ai')$

3.3.3.2 Envío de Datos

El proceso de envío de datos se realiza de la siguiente forma:

1. Creación del Objeto: Con la siguiente instrucción se realiza la creación de la salida analógica a la cual se la denominará *ao*.

$ao = analogoutput('winsound')$

2. Agregar canales: Después de haber creado el objeto se debe agregar los canales los cuales serán los caminos por donde se realizará el envío de datos. El número de canales a agregar es dos.

```
addchannel(ao,1:2)
```

3. Configurar valores de las propiedades: Después de haber agregado canales al objetos se debe configurar la velocidad de muestreo (44.1 KHz).

```
set(ao,'SampleRate',44100)
```

4. Envío de datos:

```
putdata(ao,[datos  datos])
start(ao)
```

5. Borrar: Cuando ya no se necesite el objeto es mejor borrar y limpiar, es decir liberar memoria

```
delete(ao)
clear('ao')
```

3.3.3.3 Lectura / Escritura de valores Digitales

El proceso de lectura / escritura de valores digitales se realiza de la siguiente forma:

1. Creación del Objeto: Con la siguiente instrucción se realiza la creación de la entrada / salida digital a la cual se la denominará "*dio*". *Para* este

ejemplo se utilizará la tarjeta de adquisición de datos PCI-DAS 1000, la cual tiene un subsistema de entradas / salidas digitales[13].

```
dio = digitalio('CBI',0)
```

2. Agregar líneas: Después de haber creado el objeto se debe agregar líneas las cuales serán los caminos por donde se realizará el envío / recepción de datos. El número de líneas a agregar es máximo 24, para este caso solo se configurará ocho líneas. Además se debe configurar la dirección (entrada / salida o entrada); en caso de configurar como 'out' se podrá escribir y leer valores digitales, mientras que si se configura como 'in' solo se podrá leer valores digitales.

```
addline(dio,0:7,'out')
```

3. Lectura / Escritura de valores: Para escribir un valor se lo puede hacer de dos formas, ya sea creando un vector o directamente designando un valor decimal, como ejemplo se escribirá de las dos formas.

```
putvalue(dio,[1  0  1  1  0  0  1  0])
o
putvalue(dio,178)
```

En caso que se requiera leer datos se escribe el siguiente comando:

```
getvalue(dio)
```

4. Borrar: Cuando ya no se necesite el objeto es mejor borrar y limpiar, es decir liberar memoria

---

[13] La tarjeta de sonido de Windows solo cuenta con dos subsistemas, un subsistema de entrada analógica y un subsistema de salida analógica.

delete(dio)
clear('dio')

3.3.3.4 Ejemplo de la creación de un objeto

A continuación se muestra un ejemplo para la adquisición de datos, se indica el proceso que se realiza, en lo referente a líneas de comandos, para adquirir datos:

Como se indicó en el proceso del numeral 3.3.2.1, primero se debe crear el objeto.

```
>> ai=analoginput('cbi',1)

Display Summary of Analog Input (AI) Object Using 'PCI-DAS1000'.

  Acquisition Parameters:  1000 samples per second on each channel.
                           1000 samples per trigger on each channel.
                           1 sec. of data to be logged upon START.
                           Log data to 'Memory' on trigger.

      Trigger Parameters:  1 'Immediate' trigger(s) on START.

           Engine status:  Waiting for START.
                           0 samples acquired since starting.
                           0 samples available for GETDATA.

AI object contains no channels.
```

FIGURA 3.17) Matlab: Ejemplo de la creación de una entrada analógica

Luego se debe agregar canales:

```
>> addchannel(ai,0:3,{'Canal 1' 'Canal 2' 'Canal 3' 'Canal 4'})

   Index:  ChannelName:  HwChannel:  InputRange:  SensorRange:  UnitsRange:  Units:
   1       'Canal 1'     0           [-10 10]     [-10 10]      [-10 10]     'Volts'
   2       'Canal 2'     1           [-10 10]     [-10 10]      [-10 10]     'Volts'
   3       'Canal 3'     2           [-10 10]     [-10 10]      [-10 10]     'Volts'
   4       'Canal 4'     3           [-10 10]     [-10 10]      [-10 10]     'Volts'
```

FIGURA 3.18) Matlab: Ejemplo de agregar canales a una entrada analógica

Los valores por defecto de la velocidad de muestreo (SampleRate) y de las muestras tomadas por cada vez que se realiza un muestreo (SamplesPerTrigger) de la tarjeta de adquisición de datos son 1000 Hz y 1000 muestras respectivamente. Luego se procede a configurar las dos propiedades mencionadas anteriormente.

```
>> set(ai,'SampleRate',10)
>> set(ai,'SamplesPerTrigger',5)
```

FIGURA 3.19) Matlab: Configurar la velocidad de muestreo y las muestras que se toman en cada canal de una entrada analógica

Después de haber configurado el objeto se procede a empezar a tomar las muestras:

```
>> start(ai)
```

FIGURA 3.20) Matlab: Comenzar a adquirir datos con la entrada analógica

Después de haber tomado las muestras se la puede asignar a una variable para poder realizar la manipulación de los datos adquiridos.

```
>> [Amplitud,tiempo]=getdata(ai,ai.SamplesPerTrigger)

Amplitud =

    0.5788    0.6422    0.6227    0.5983
    0.5836    0.6764    0.6569    0.6032
    0.5788    0.6911    0.6862    0.6178
    0.5690    0.6813    0.6911    0.6276
    0.5641    0.6764    0.6911    0.6276


tiempo =

         0
    0.1000
    0.2000
    0.3000
    0.4000
```

FIGURA 3.21) Matlab: Muestras adquiridas con la entrada analógica

Los datos adquiridos se los puede graficar, realizar diferentes operaciones matemáticas; es decir, se puede manipular dichos datos. Como por ejemplo se graficará los datos adquiridos:

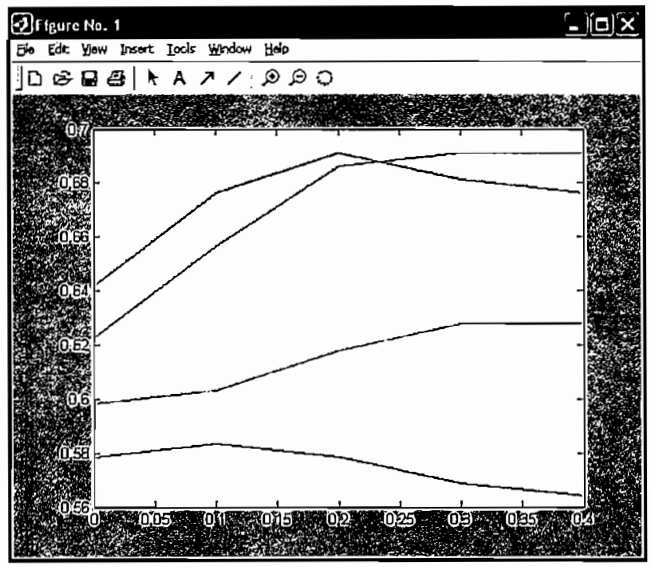


FIGURA 3.22) Matlab: Gráfico de las muestras adquiridas con la entrada analógica

Para obtener información de un objeto, como puede ser una entrada analógica, una salida analógica o una entrada / salida digital, primero se debe crear los objeto con las instrucciones que se mostró anteriormente, luego de haber creado un determinado objeto se procede a escribir la siguiente línea de comando:

```
>> Informacion=daqhwinfo(ai)

Informacion =

                AdaptorName: 'cbi'
                       Bits: 12
                   Coupling: {'DC Coupled'}
                 DeviceName: 'PCI-DAS1000'
            DifferentialIDs: [0 1 2 3 4 5 6 7]
                      Gains: []
                         ID: '1'
                 InputRanges: [8x2 double]
              MaxSampleRate: 250000
              MinSampleRate: 1
             NativeDataType: 'uint16'
                   Polarity: {'Bipolar'}
                 SampleType: 'Scanning'
             SingleEndedIDs: [0 1 2 3 4 5 6 7]
              SubsystemType: 'AnalogInput'
               TotalChannels: 8
    VendorDriverDescription: 'Computer Boards Universal Library'
       VendorDriverVersion: '5'
```

FIGURA 3.23) Matlab: Información de las propiedades de la entrada analógica

```
>> voltaje=Informacion.InputRanges

voltaje =

         0    1.2500
         0    2.5000
         0    5.0000
         0   10.0000
   -1.2500    1.2500
   -2.5000    2.5000
   -5.0000    5.0000
  -10.0000   10.0000
```

FIGURA 3.24) Matlab: Información de los voltaje de entrada máximos permitidos por la DAQ

En el capítulo anterior se habló de los subsistemas, los cuales conforman la tarjeta de adquisición de datos, a continuación se muestra la relación entre los subsistemas y los objetos de la herramienta de adquisición de datos.

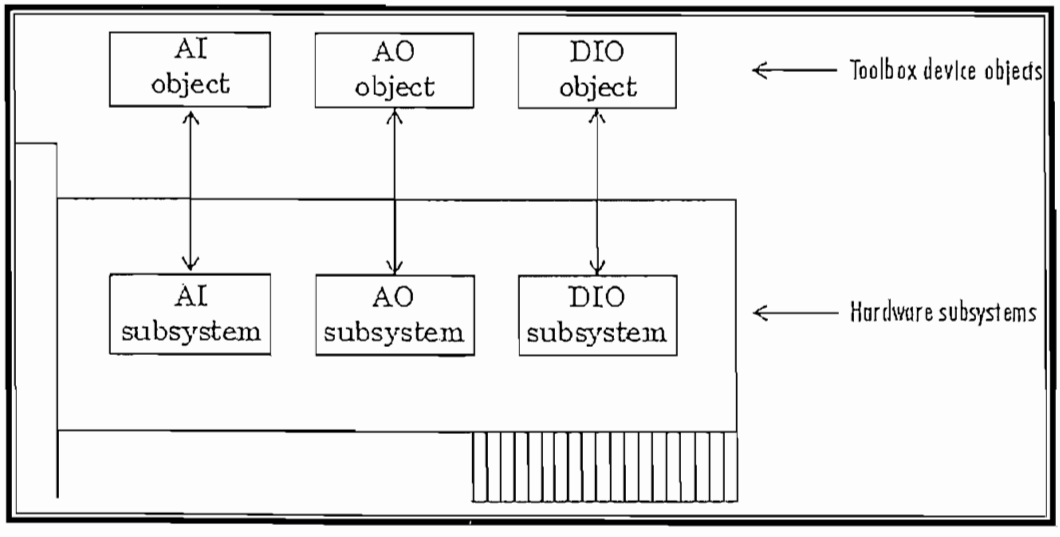


FIGURA 3.25) Comunicación entre los subsistemas y los objetos de la herramienta de Adquisición de Datos

3.3.4 CREACIÓN DE INTERFAZ DE USUARIO

Para crear un interfaz gráfico de usuario en MATLAB se puede hacer de dos formas:

3.3.4.1 Creación de forma gráfica

La primera forma es realizar gráficamente el interfaz, digitando el comando *guide* se muestra una pantalla en la cual se puede agregar diferentes controles.

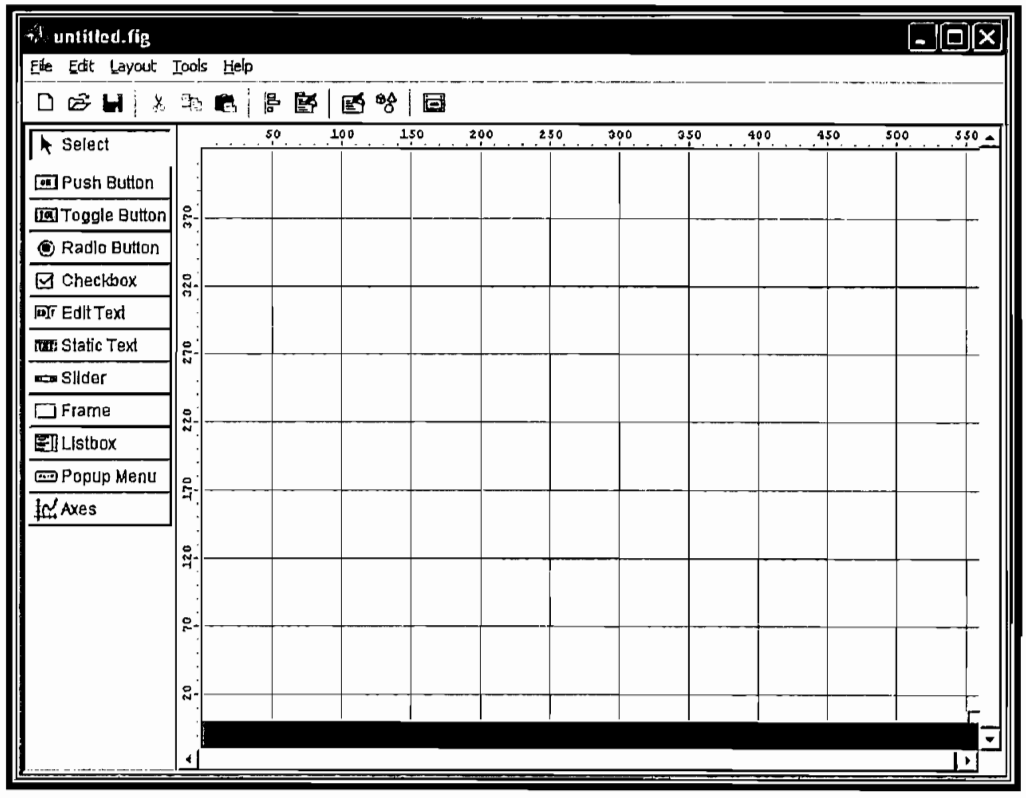


FIGURA 3.26) Matlab: Creación de interfaces de usuario con el comando "GUIDE"

3.3.4.1.1 Controles para interfaz de usuario

Los controles sirven para ejecutar una o varias acciones[14] en el interfaz gráfico de usuario, los controles que se puede tener en un interfaz de usuario son:

- ✓ Push buttons
- ✓ Toggle button
- ✓ Radio button
- ✓ Checkbox

[14] Las acciones se relaciones con las funciones, las cuales son almacenadas en archivos .m

- ✓ Edit text
- ✓ Static Text
- ✓ Slider
- ✓ Frame
- ✓ List box
- ✓ Popup menu
- ✓ Axes

1 Push Button

Este control ejecuta una acción cuando es presionado. Para ejecutar alguna acción predefinida en este control se debe hacer un clic sobre dicho botón, después de realizarse el mencionado clic el botón vuelve a su posición original (botón sin presionarse).

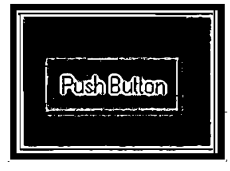


FIGURA 3.27) Matlab: representación gráfica de Push Button

2 Toggle Button

Este control ejecuta una acción cuando es presionado. A diferencia del control *push button* al hacer un clic sobre el botón este se mantiene presionado ejecutando una acción predefinida hasta que se vuelva a hacer un clic para que vuelva a su estado original y se deje de ejecutar cualquier acción.

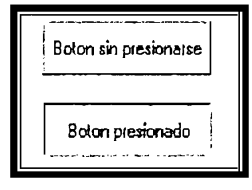


FIGURA 3.28) Matlab: representación gráfica de Toggle Button

3 Radio Button

Este control sirve para ejecutar solo una acción de un grupo de acciones.

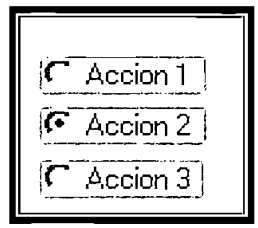


FIGURA 3.29) Matlab: representación gráfica de Radio Button

En la figura anterior se muestra un ejemplo del uso de los controles *radio button*, en los cuales del grupo de acciones (Acción 1, Acción 2 y Acción 3) solo se ejecuta una acción (Acción 2).

4 CheckBox

Este control sirve para ejecutar una, varias o todas las acciones de un grupo de acciones.

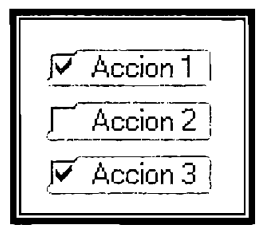


FIGURA 3.30) Matlab: representación gráfica de CheckBox

En la figura anterior se muestra un ejemplo del uso de los controles *CheckBox*, en los cuales del grupo de acciones (Acción 1, Acción 2 y Acción 3) se ejecutan dos acciones (Acción 1 y Acción 3).

5 Edit Text

Con este control el usuario puede ingresar información de forma escrita al programa, como por ejemplo si se requiere realizar una suma definida por el usuario, el usuario puede ingresar los valores a sumar (5 y 2,28).

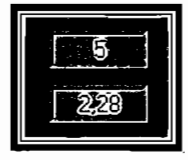


FIGURA 3.31) Matlab: representación gráfica de Edit Text

6 Static Text

Este control se utiliza para que el sistema muestre información al usuario, en referencia al ejemplo anterior este control servirá para mostrar el valor de la suma al usuario.

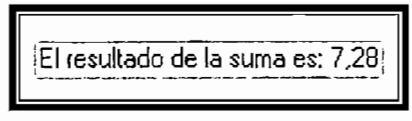


FIGURA 3.32) Matlab: representación gráfica de Static Text

7 Slider

Este control se muestra gráficamente como una barra que se desplaza tomando diferentes valores; es decir, que el valor de esta barra depende de la posición en que se encuentre.

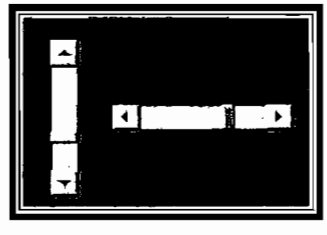


FIGURA 3.33) Matlab: representación gráfica de Slider

8 *Frame*

Este control se utiliza para agrupar otros controles.

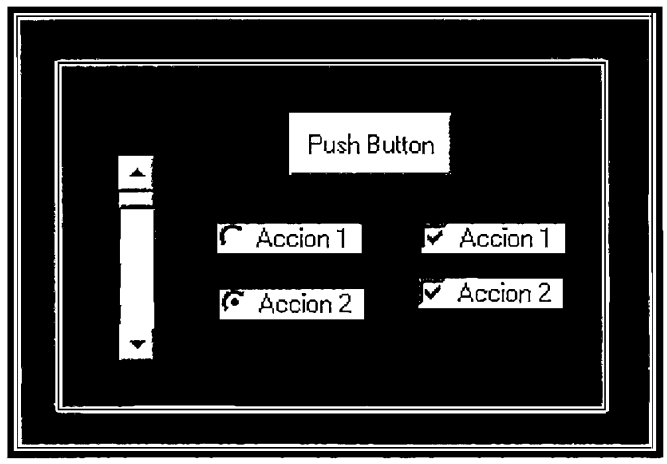


FIGURA 3.34) Matlab: representación gráfica de Frame, frame contiene varios controles mencionados anteriormente

9 *List Box*

Este control se utiliza para seleccionar una acción de una lista de acciones, en el ejemplo siguiente se muestra lo siguiente: de la lista de acciones (Acción 1, Acción 2, Acción 3 y Acción 4) se selecciona realizar la tercera acción (Acción 3).

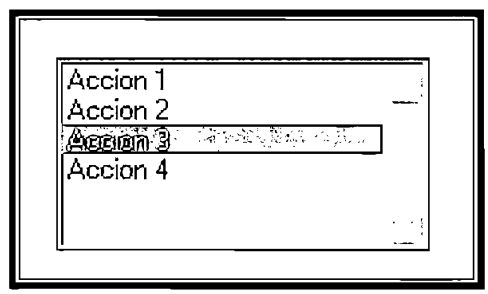


FIGURA 3.35) Matlab: representación gráfica de List Box

10 Popup menu

Este control tiene una funcionalidad semejante al control anterior (List Box), se diferencia del control anterior cómo se muestra gráficamente, como ejemplo se ha seleccionado que realice la Acción 2.

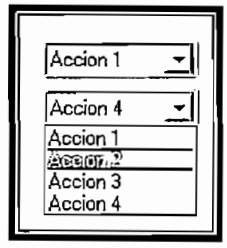


FIGURA 3.36) Matlab: representación gráfica de Popup Menu

11 Axes

Este control sirve para mostrar gráficamente datos que se han obtenido de una adquisición de datos o graficar una señal que se ingrese mediante software.

Como ejemplo de lo descrito en el párrafo anterior a continuación se muestra una gráfica de una señal aleatoria.

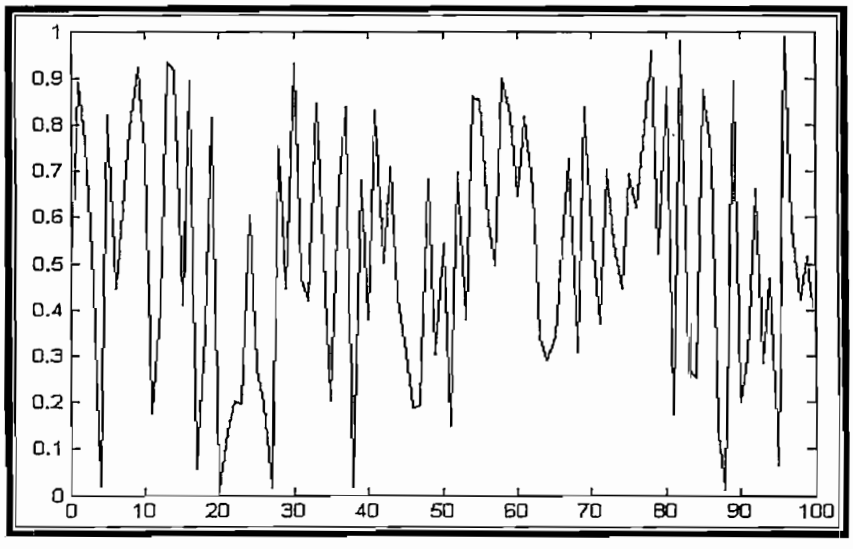


FIGURA 3.37) Matlab: representación gráfica de una señal aleatoria

3.3.4.1.2 Propiedades de los controles

Los controles tienen propiedades que pueden ser configuradas, entre las propiedades más importantes están:

1 BackGoundColor

Esta propiedad sirve para configurar el color que el usuario desea que tenga el control.

2 CallBack

Esta propiedad sirve para que cuando se realice alguna acción (por ejemplo un clic) sobre el control este ejecute un programa (acción almacenada en archivos *.m).

3 Enable

Esta propiedad sirve para habilitar o deshabilitar un control; es decir, que el usuario pueda o no acceder a ejecutar alguna acción sobre un control.

4 Font Name

Con esta propiedad se puede seleccionar el tipo de letra que se desea que tenga el control, esto es cuando el control cuente con letras como es el caso de un *push button* o un *static text*, pero no es el caso de un *slider.*

5 Font Size

Con esta propiedad se configura el tamaño de la letra.

6 ForegroundColor

Sirve para configurar el color de las letras.

7 HorizontalAlignment

Sirve para configurar la alineación de las letras, esto puede ser alineado a la izquierda, a la derecha o al centro.

8 Position

Sirve para configurar la posición en la que se va a encontrar el control dentro de la pantalla.

9 String

Es el texto del control que se va a mostrar al usuario

10 Style

Indica el tipo de control, como por ejemplo si es *toggle button*, *checkbox*, etc.

11 Visible

Indica si el control es visible o no para el usuario.

También existen otras propiedades, las cuales se las mencionará conforme se indique como se realizó la programación.

A continuación se muestra como generar una interfaz gráfica de usuario agregando los controles mencionados en los párrafos anteriores.

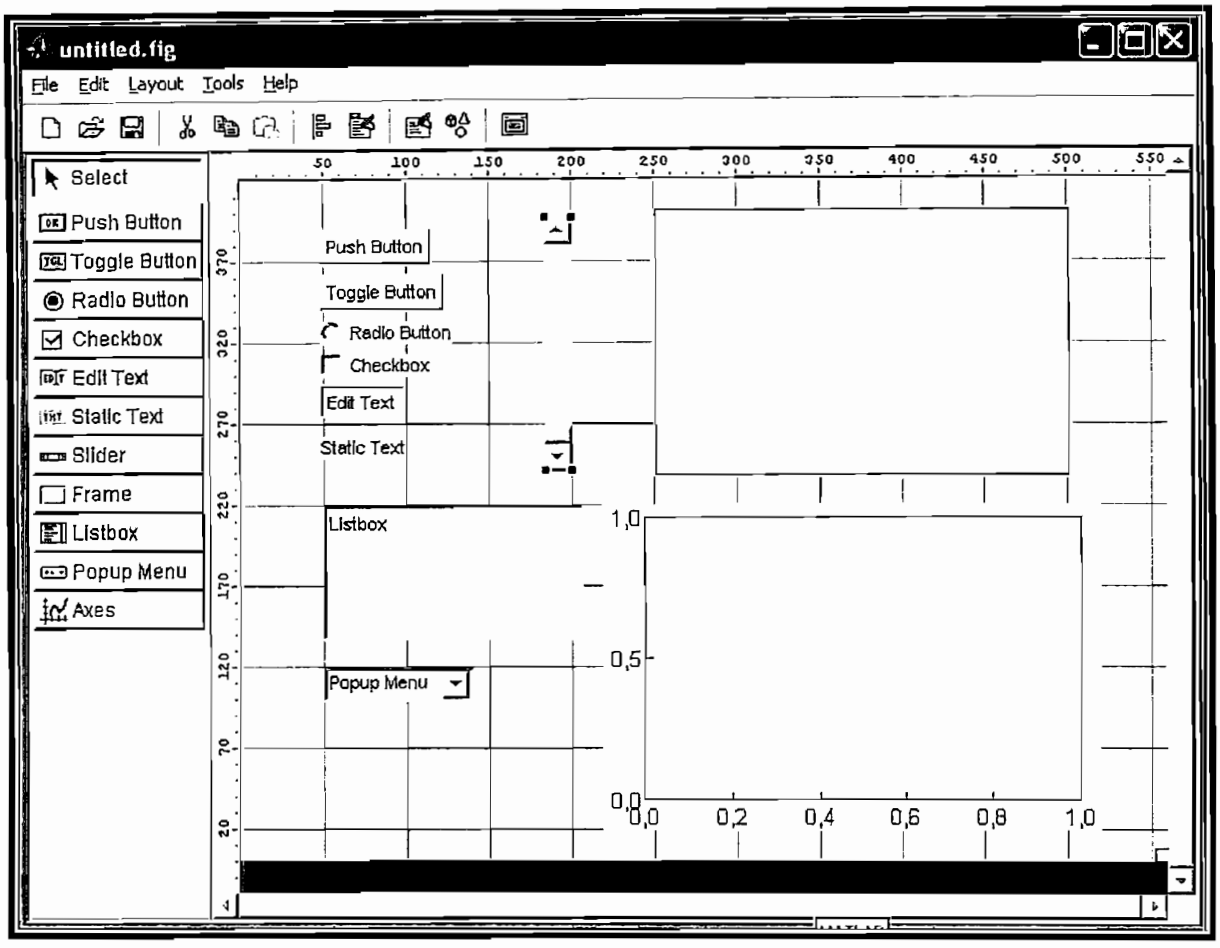


FIGURA 3.38) Matlab: representación varios controles en la pantalla para crear interfaces de usuario

3.3.4.2 Creación por código

La segunda forma de crear interfaces gráficos de usuario es mediante código, esta es la forma como se realizó la programación del interfaz de usuario para este proyecto.

Primero se debe crear la figura (*figure*) donde se encontraran los controles. Para esto se debe escribir el siguiente código, el cual estará almacenado en un archivo de extensión .m:

```
figure('Units','normalized',...
        'Color',[1 1 1],...
        'Menubar','none',...
        'Name','Pantalla 1',...
        'NumberTitle','off',...
        'Position',[0 0 1 1],...
        'ToolBar','none',...
        'Resize','on',...
        'Visible','on');
```

Donde Units, Color, Menubar, etc. son las propiedades que caracterizarán la pantalla donde se encuentren los controles, como por ejemplo el código anterior indica: Creación de una pantalla (figura) donde la unidades de la pantalla están normalizadas[15]. Además indica el color que va a tener la pantalla, no va a tener barra de menús, el nombre de la pantalla es "Pantalla 1", la pantalla no va a estar numerada ya que MATLAB numera las pantallas que grafica de forma secuencial, además indica la posición que va a ocupar la pantalla en el monitor del computador, no va a contar con barra de herramientas, va a poder cambiar de tamaño y va a ser visible al usuario.

Al realizar el desarrollo del interfaz se va a profundizar más en las propiedades específicas que se esté usando.

Después de haber creado la pantalla se debe continuar creando los controles, el código usado para crear los diferentes controles es semejante, lo que varia es en las propiedades de cada uno de ellos, a continuación se indicará un ejemplo de cómo crear un *static text*.

```
uicontrol('Units',normalized,...
        'BackgroundColor',[1 0 1],...
```

[15] Unidades Normalizadas: mínimo valor 0 y máximo valor 1

```
'FontSize',18,...
'ForegroundColor','[0 1 1]',...
'Horizontalalignment','center',...
'Position',[0.15 0.6 0.7 0.3],...
'String','Ejemplo para crear un control',...
'Style','text');
```

Donde Style, String, Position, etc. son las propiedades que caracterizarán el control, como por ejemplo el código anterior indica: Creación de un control donde la unidades estan normalizadas; indica el color que va a tener el control, el tamaño de la letra, el color de las letras; el alineamiento de las letras (las letras van a estar centradas respecto al control), la posición que va a ocupar en la pantalla, las letras que va a tener el control, en este caso va a indicar "Ejemplo para crear un control" e indica el estilo; es decir, indica el tipo de control.

3.4 ESPECIFICACIÓN DE REQUERIMIENTOS

La interfaz a desarrollarse debe permitir lo siguiente:

- Manipular nivel de Entrada de las señales (micrófonos, auxiliares)
- Manipular nivel de Salida del amplificador.
- Indicar la potencia de Salida.
- Indicar el valor de la Resistencia del parlante.
- Mostrar gráficamente la respuesta de frecuencia del amplificador.
- Respuesta de un canal del amplificador introduciendo una señal a otro canal de dicho amplificador.

- Desplegar un ecualizador gráfico de tres bandas

3.5 DISEÑO

El interfaz de usuario debe contar con cuatro elementos para manipular los valores de las entradas, esto se refleja en el interfaz de usuario como cuatro barras que variando la posición actúan sobre las entradas / salidas digitales enviando pulsos hacia los potenciómetros digitales y de esta forma se varían los niveles de la señal adquirida por las entradas analógicas.

Además el interfaz cuenta con otras dos barras para manipular los valores de las salidas analógicas; es decir, se manipula la amplitud de voltaje que se envía hacia el amplificador de potencia de audio.

Se debe contar también con dos ejes en los cuales se grafica las distintas señales que se tiene ya sean señales de entrada como las señales de salida, cabe recalcar que las gráficas de las diferentes señales puede ser en el dominio del tiempo como en el dominio de la frecuencia.

Para poder realizar las acciones indicadas en el párrafo anterior se cuenta con cuatro controles (*ComboBox*) los cuales, como se explicó anteriormente, son barras de texto desplegables que permiten escoger uno de sus elementos que se encuentran en la lista, con los dos primeros *ComboBox* se escoge los puntos de donde se desea tomar las señales a graficar; es decir, si se desea tomar la señal de cualquiera de las entradas y/o de cualquiera de las salidas, mientras que los dos últimos se elige el dominio que se desea graficar las salidas que se han escogido en los dos primeros *ComboBox.*

Además el interfaz debe tener cuadros de texto los cuales indiquen los valores de potencia de salida del amplificador y valor de la impedancia del parlante.

El interfaz también tiene una barra de menús la cual permite acceder a ciertas funciones especiales del software como son: cerrar la aplicación, inhabilitar ciertos controles, llamar a una pantalla la cual permita realizar la función de ecualización, detener la adquisición de datos y ayuda, se debe programar de tal forma que se pueda tener accesos a estas funciones con teclas de función como por ejemplo: para ejecutar la acción de cerrar la aplicación se debería presionar las teclas ctrl. + Q, se utiliza la letra Q para que junto con la tecla ctrl. cierren la aplicación ya que en el software MatLab no se puede usar las letras "C", "X" o "V" para usarlas como funciones especiales ya que por valores de defecto del sistema esas letras están destinadas a Copiar, Cortar y Pegar respectivamente (Copiar = ctrl. + C, Cortar = ctrl. + X, Pegar = ctrl. + V)

3.6 IMPLEMENTACIÓN

El software desarrollado para el control del amplificador de potencia de audio muestra una pantalla inicial o de bienvenida denominada *Inicio* (figura 3.39).

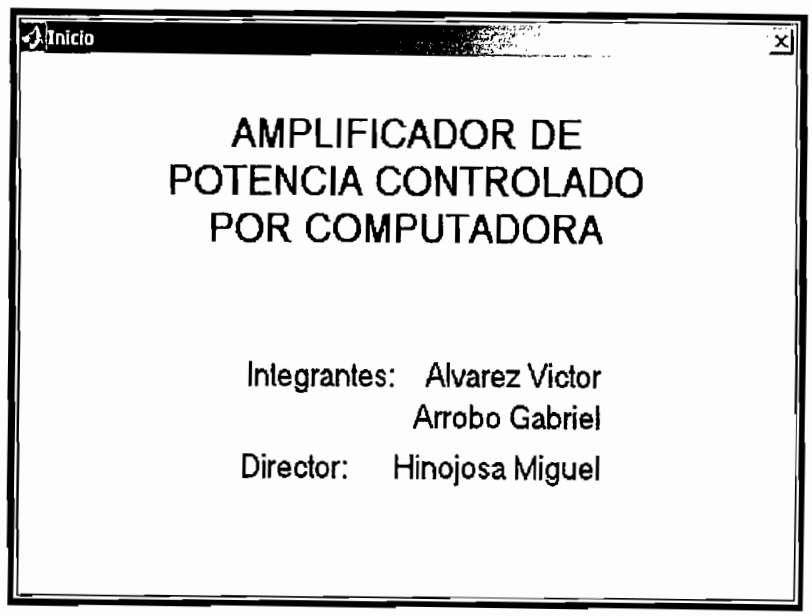


FIGURA 3.39) Matlab: programa de Inicio del controlador

Las líneas de comando para poder graficar la pantalla mostrada en la figura anterior son:

```
figini=figure(...
    'Units',uni,...
    'Color',colorfondo,...
    'Menubar','none',...
    'Name','Inicio',...
    'NumberTitle','off',...
    'Position',posicionfig,...
    'ToolBar','none',...
    'Resize','off',...
    'Visible','off',...
    'WindowStyle','Modal');
```

donde: figini es el nombre de la variable asignada a la figura

figure sirve para crear un objeto gráfico, en este caso la pantalla (figura) con sus respectivas propiedades que se describen a continuación.

'Units' son la unidades de la pantalla, quiere decir si la escala para graficar va a estar en pulgadas, centímetros, pixels o en un valor normalizado

'Color' es el color de la figura

'Menubar' es para tener en la pantalla una barra de menús

'Name' es el nombre de la pantalla, en este caso "Inicio"

'NumberTitle' es el número de pantalla que se grafica con MatLab

'Position' es la ubicación en la que se grafica esta pantalla en el monitor del computador.

'ToolBar' es para tener en la pantalla una barra de herramientas

'Resize' es para cambiar de tamaño a la pantalla

'Visible' es para que la pantalla pueda ser vista en el monitor.

'WindowStyle' es para permitir o no minimizar la pantalla "Inicio"

La pantalla *Inicio* contiene tres cuadros de texto, los cuales indican el nombre del proyecto, el nombre de las personas que desarrollaron este proyecto y el nombre del director del proyecto, los tres cuadros de texto tienen las mismas propiedades, lo único en que difieren es el valor de ciertas propiedades que los conforman, a continuación se indicará el código para crear uno de estos cuadros de texto.

```
uicontrol(...
        'Units',uni,...
        'BackgroundColor',colorfondo,...
        'FontSize',18,...
        'FontWeight','bold',...
        'ForegroundColor','b',...
        'Horizontalalignment','center',...
        'Position',[0.15 0.6 0.7 0.3],...
        'String','AMPLIFICADOR DE POTENCIA CONTROLADO POR
        COMPUTADORA',...
        'Style','text');
```

donde: uicontrol sirve para crear un objeto (en este caso un cuadro de texto), este es un control que está conformado por varias propiedades

'BackgrondColor' es para configurar el color del fondo (relleno) del cuadro de texto

'FontSize' es para configurar el tamaño de la letra que contiene el cuadro de texto

'FontWeight' es para configurar el grosor de las letras, este se puede configurar como: light, normal (valor por defecto), demi o bold, para esta propiedad se puede hacer una analogía con Microsoft Windows el cual permite como 'FontWeight' a las letras: normal o negrilla

'ForegroundColor' es para configurar el color de las letras que se encuentran en el cuadro de texto

'Position' es para configurar la posición en la que se va a ubicar el cuadro de texto dentro de la figura "Figini"
'String' es el texto que va a tener el cuadro de texto
'Style' es para definir que tipo de control se tiene, en este caso es un cuadro de texto. El campo 'Style' puede tener los siguientes valores: pushbutton, togglebutton, radiobutton, checkbox, edit, text, slider, frame, listbox, popupmenu, por defecto el valor 'Style' de los objetos uicontrol es pushbutton

Para lo otros dos cuadros de texto que conforman la pantalla *Inicio* la configuración es similar y como se mencionó anteriormente lo que cambia es el valor que tienen algunas propiedades.

La pantalla *Inicio* está programada de tal forma que después de 5 segundos estar ejecutada se cierra automáticamente y hace el llamado a la figura para la selección de los subsistemas denominada *GrafCONF*.

El código para poder realizar la acción descrita anteriormente es el siguiente:

pause(5);	Detiene la aplicación durante 5 segundos
close(gcf);	Cierra la pantalla "Inicio" (pantalla ejecutándose en ese momento)
GrafCONF;	Llama al programa *GrafCONF*

El programa *GrafCONF* permite seleccionar los diferentes subsistemas (entradas analógicas, salidas analógicas y entradas / salidas digitales), para realizar el control, con los que cuentan las diferentes tarjetas de adquisición de datos.

La programación de esta se inicia definiendo la variable "Datos", la cual es una estructura que almacena toda la información; es decir, almacena la información de las entradas analógicas con los valores adquiridos, entradas /

salidas digitales, los elementos de la pantalla de usuario con sus respectivos valores, etc.

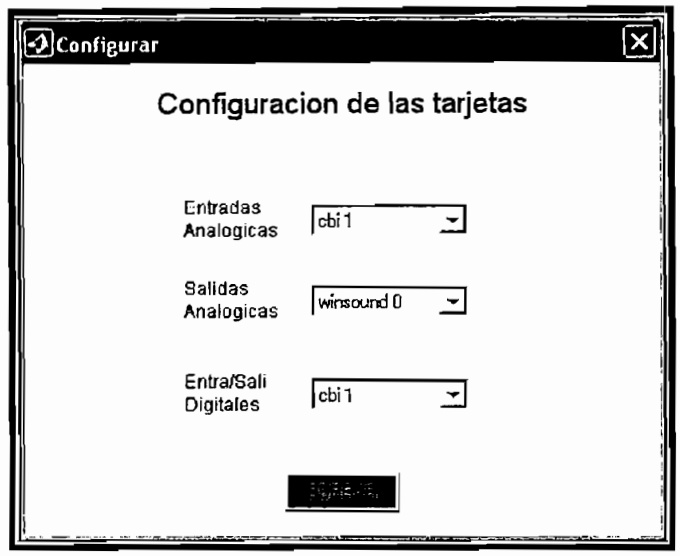


FIGURA 3.40) Matlab: programa para configurar los subsistemas a utilizar

En la pantalla *GrafCONF*[16] (figura 3.40) se selecciona los subsistemas a utilizar para la adquisición y envío de información. En el *ComboBox* de *Entradas Analógicas* se selecciona la tarjeta que contenga el subsistema que se utilizará para la adquisición de datos, en el *ComboBox* de *Salidas Digitales* de igual forma se selecciona la tarjeta que contenga el subsistema que se utilizará para el envío de datos, mientras que en el ComboBox de Entra/Sali Digitales se selecciona la tarjeta que contenga el subsistema para la manipulación de datos digitales; es decir, el programa busca todas las tarjetas de adquisición de datos que contenga el computador separándolas por subsistemas, por ejemplo para el caso de las Entradas Analógicas el programa busca todas las tarjetas que adquisición de datos que se encuentren en el computador instaladas y configuradas que cuenten con un subsistema que sea *analoginput (Entrada Analógica)* y de esta manera poder seleccionar con la tarjeta que se desee trabajar (figura 3.41). De igual forma se procede para la selección de los otros dos subsistemas (*Salidas Analógicas, Entradas / Salidas Digitales*).

---

[16] El código para realizar todo lo que se describe acerca de la pantalla GrafCONF se indica en el anexo B.

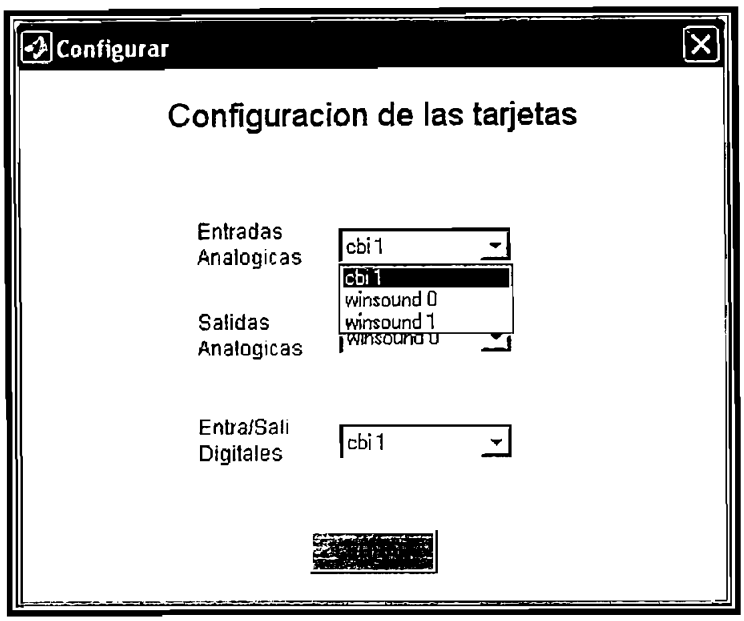


FIGURA 3.41) Matlab: Ejemplo de selección de subsistema de *Entradas Analógicas*

En el capítulo 2 se detalla los requerimientos con respecto al números de canales analógicos y al número de líneas digitales (*10 entradas analógicas, 2 salidas analógicas y 8 entradas / salidas digitales*), en caso que el usuario seleccione algún subsistema que no cumpla con los requerimientos planteados en el capítulo 2, el programa mostrará el mensaje respectivo referente al error cometido. Por ejemplo en caso de seleccionar la tarjeta de sonido *WinSound*[17] para la adquisición de datos el mensaje que se muestra es el indicado en la figura 3.42.

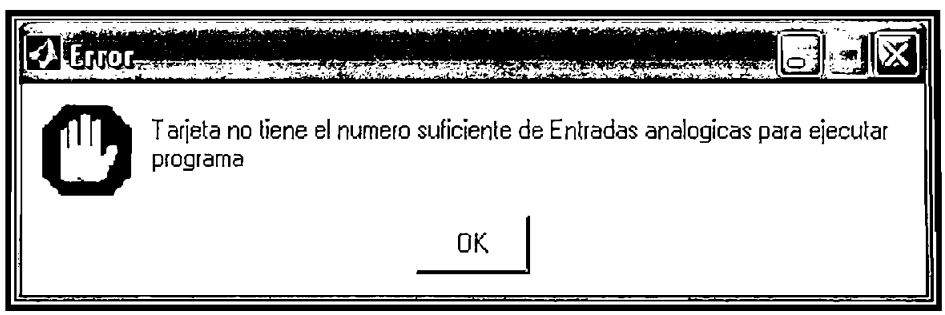


FIGURA 3.42) Matlab: Mensaje de error en la selección de subsistema de *Entradas Analógicas*

[17] La tarjeta de Sonido *WinSound* cuenta con dos subsistemas (Entradas Analógicas y salidas Analógicas) y cada subsistema cuenta con dos canales.

Si el usuario hace caso omiso al mensaje de error mostrado, selección los demás subsistemas y luego presiona el botón Cerrar, el programa muestra un mensaje de error de configuración, el cual le permite seleccionar nuevamente de forma adecuada los subsistemas a utilizar en caso de presionar *si*, mientras que en caso de presionar *no*, el programa termina su ejecución, figura 3.43.

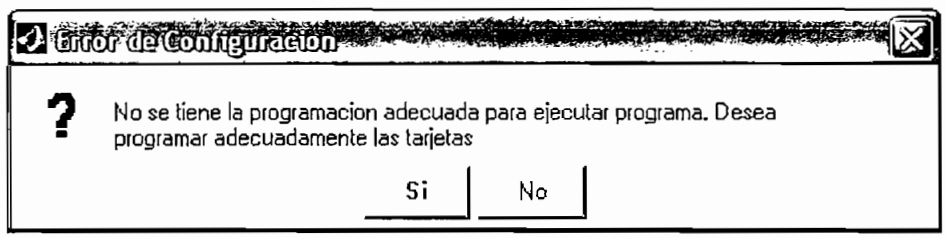


FIGURA 3.43) Matlab: Mensaje de error en la configuración

Para poder continuar a la pantalla en la cual se va a realizar el control del sistema Amplificador de Audio Controlado por Computadora se debe presionar el botón *Cerrar*, en caso de no haber seleccionado los subsistemas a utilizar el programa mostrará un mensaje de advertencia indicando que se seleccione los subsistemas a utilizar, figura 3.44.

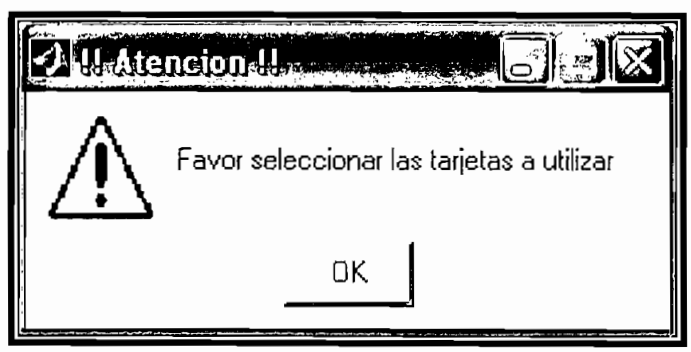


FIGURA 3.44) Matlab: Mensaje de advertencia por la no selección de los diferentes subsistemas

Después que el usuario seleccione adecuadamente los subsistemas a usar y luego de haber presionado el botón Cerrar, se hace el llamado al programa *Controlador*.

El programa *Controlador* realiza las siguientes acciones:

- Crear las entradas y las salidas analógicas
- Crear las entradas / salidas digitales
- Crear la pantalla de usuario

3.6.1 CREACIÓN DE LAS ENTRADAS Y SALIDAS ANALÓGICAS

Para crear la entrada analógica se escribe los siguientes comandos:

```
ai=analoginput('cbi',1)
```

donde: "ai" es una variable donde se almacena la entrada analógica, se usa el nombre de la variale "ai" para tener relación con el nombre del objeto, el cual es analoginput
"cbi" es el nombre que asigna MatLab 6.0 a la tarjeta de adquisición de datos (ComputerBoards)
"1" indica el # del slot PCI que ocupa la tarjeta en el computador

Luego de haber creado la entrada analógica se debe agregar canales, con los cuales se realiza la adquisición de datos.

```
addchannel(ai,0:9)
```

donde: "addchannel" es el código que se usa para agregar canales a una entrada analógica
"0:9" indica cuantos canales se requiere agregar (en este caso 10 canales, del 0 al 9)

```
ao=analogoutput('winsound')
```

donde: "ao" es una variable donde se almacena la salida analógica, se usa el nombre de la variale "ao" para tener relación con el nombre del objeto, el cual es analogoutput
"winsound" es el nombre que asigna MatLab 6.0 a la tarjeta de sonido

Luego de haber creado la salida analógica se debe agregar canales, con los cuales se realiza la adquisición de datos.

```
addchannel(ao,1:2)
```

donde: "addchannel" es el código que se usa para agregar canales a una entrada analógica
"1:2" indica cuantos canales se requiere agregar (en este caso 2 canales)

En algunas tarjetas de adquisición de datos se puede configurar las entradas analógicas como: entradas simples o entradas diferenciales, esto se lo realiza con la siguiente línea de programa:

```
set(ai,'InputType','Single')
```

Para el caso de las salidas analógicas, la instrucción siguiente indica que no se va a trabajar con velocidades de muestreo estandarizadas como lo son por ejemplo: 8000, 11025, 22050, 44100, sino se va a trabajar con una velocidad diferente.

```
set(ao,'StandardSampleRates','Off');
```

Se debe programar la velocidad de muestreo de los canales, para encontrar el valor máximo de velocidad de muestreo de cada canal se debe conocer la

velocidad de muestreo máxima de la DAQ (en el caso de la tarjeta con la que se está trabajando en este proyecto es 250 KHz)

Para el caso que se configure 10 canales de entrada la velocidad máxima de cada canal es:

$$V_{CANAL} = \frac{V_{MAX}}{\#\ canales}$$
$$V_{CANAL} = \frac{250000}{10}$$
$$V_{CANAL} = 25000$$

Ecuación 3.1 Calculo de la velocidad máxima de muestreo por canal

```
set(ai,'SampleRate',25000);
set(ao,'SampleRate',25000);
```

Después de realizar la configuración de la velocidad de muestreo se procede a configurar el número de muestras a tomar en cada muestreo

```
set(ai,'SamplesPerTrigger',inf);
```

Con la instrucción mencionada anteriormente se indica a la entrada analógica que debe realizar un muestreo continuo.

Con la siguientes instrucciones se indica que cada vez que se envíe 2500 valores al subsistema de Salida Analógica se ejecuta una acción, la cual es llamar llamar a la función *actualizar*, la cual se encuentra almacenada en el archivo *actualizar.m*.

```
datos.ai=ai
datos.ao=ao
pru=datos.ai;
set(datos.ao,'SamplesOutputAction',{'actualizar',pru});
datos.muestras=2500;
```

```
set(datos.ao,'SamplesOutputActionCount',datos.muestras);
```

Con la siguiente línea de comandos se configura el programa para que muestre un mensaje cuando los subsistemas entrada analógico y salida analógico se detengan; es decir dejen de adquirir o enviar datos

```
set([datos.ai datos.ao],'StopAction','daqaction');
```

La instrucción que se indica a continuación inicia la adquisición de datos

```
start([datos.ai datos.ao]);
```

3.6.2 CREACIÓN DE LAS ENTRADAS / SALIDAS DIGITALES

Para crear las entradas / salidas digitales se escribe los siguientes comandos:

```
dio=digitalio('cbi',1)
```

donde: "dio" es una variable donde se almacena las entradas / salidas digitales
"cbi" es el nombre de la tarjeta de adquisición de datos
1 indica el # del slot PCI que ocupa la tarjeta en el computador

Luego de haber creado la entrada / salida digital se debe agregar líneas, las cuales se las utiliza para tomar datos digitales o enviar datos digitales a algún elemento externo de la tarjeta de adquisición de datos.

Lo mencionado en el párrafo anterior se lo realiza escribiendo la siguiente línea:

```
addline(dio,0:6,'Out');
```

donde: dio es la variable donde se almacena toda la información concerniente a la adquisición digital
0:6 es el número de líneas que va a tener la variable "dio" para la adquisición; es decir, se va a tener 8 líneas para tomar datos o enviar datos digitales (0L o 1L)
'Out' indica que las líneas son bidireccionales (IN y OUT)

3.6.3 CREACIÓN DE LA PANTALLA DE USUARIO

Para crear la pantalla de usuario se escribe varias líneas (ver Anexo 3.1), las cuales son para crear los diferentes elementos con los que cuenta la pantalla de usuario como son: cuadros de texto, ejes para gráficos, barra de menús de la pantalla, botones, etc.

La pantalla que observa el usuario después de ejecutar las diferentes líneas de comando es la siguiente:

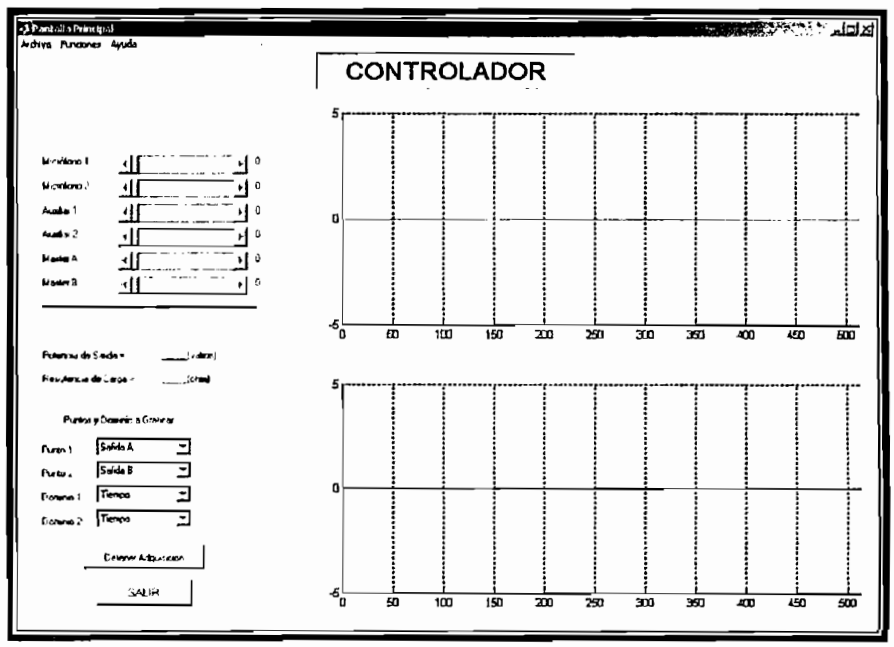


FIGURA 3.45) Matlab: programa Controlador del Amplificador de Potencia

Como se mencionó anteriormente, el sistema cuenta con un ecualizador de tres bandas, a continuación se muestra la pantalla del ecualizador, figura 3.46.

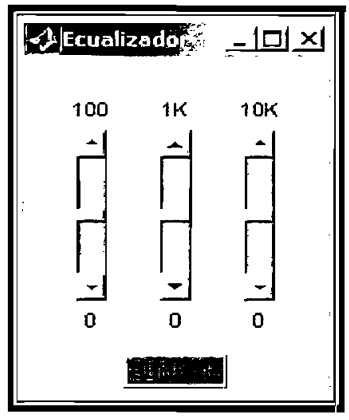


FIGURA 3.46) Matlab: Gráfica del ecualizador

Para realizar el diseño de los filtros pasabanda que conforman el ecualizador se lo realiza con una herramienta de MatLab denominada *Filter Design ToolBox*, para ejecutar esta herramienta se debe ejecutar el comando FDATool (figura 3.47)

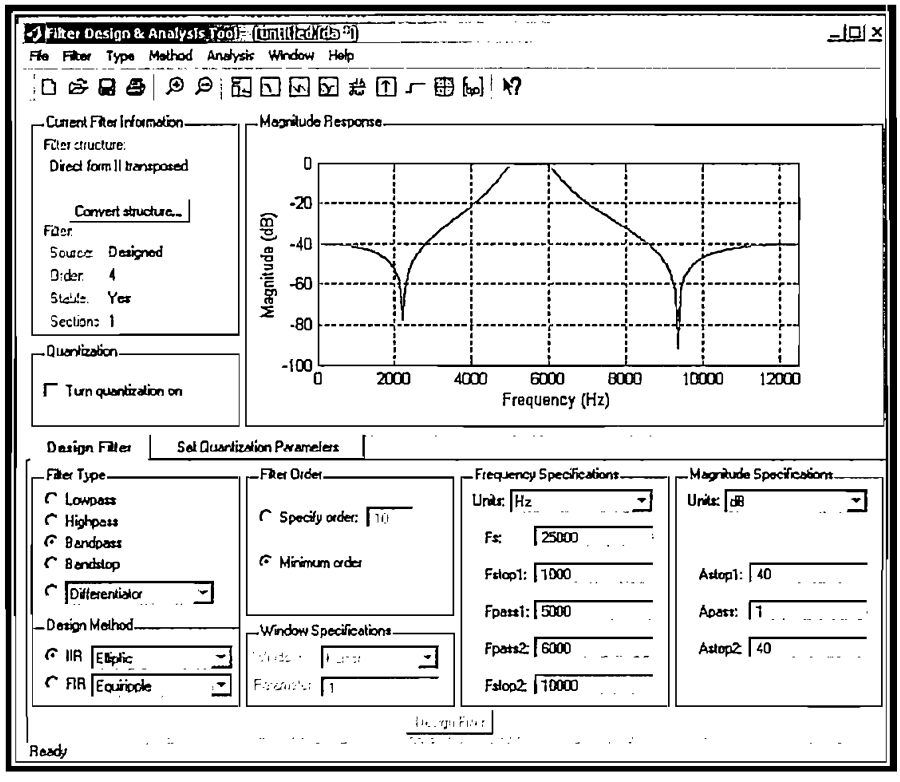


FIGURA 3.47) Matlab: FDATool herramienta para el diseño de filtros

Para el diseño de filtros pasabanda, en este caso, filtros elípticos, se debe configurar de la siguiente forma:

$[b,a] = ellip(n, A_{pass}, A_{stop}, [F_{low}\ F_{high}]/(Fs/2))$

Donde: b y a son los coeficientes del filtro

N es el orden del filtro

A_{pass} ganancia del rango de frecuencias que pasan (no son rechazados por el filtro)

A_{stop}, atenuación de las frecuencias que se encuetran fuera del filtro, es decir, frecuencias rechazadas por el filtro

F_{low} frecuencia inferior

F_{high} frecuencia superior

Fs frecuencia demmuestreo

Todo software debe estar acompañado por ayudas, para de esta forma hacer más fácil su utilización al usuario, el software desarrollado cuenta con ayudas, las cuales son: ayuda del funcionamiento, ayuda acerca de conceptos, ayuda acerca de código y otras.

3.7 PRUEBAS

Las pruebas realizadas son las de utilización del software por parte de diferentes usuarios, los usuarios han sido: un Ingeniero en electrónica y telecomunicaciones; una persona con muchos años de experiencia en sonido en vivo y un individuo que no tiene relación alguna con lo referente al sonido.

Cada una de las personas mencionadas en el párrafo anterior han dado sus opiniones:

"Es un software amigable para el usuario, no es muy complicado y además cuando se produce un error el sistema informa de lo acontecido"

"El software permite al usuario tener conocimiento de lo que está sucediendo con respecto a las entradas de los preamplificadores y las salidas del amplificador de potencia, es una ayuda adicional a la experiencia obtenida durante toda la carrera en el sonido en vivo"

"El software no le pareció tan sencillo pero con los archivos de ayuda con los que cuenta el software permitió continuar con el uso pero con un poco de dificultad"

3.8 UTILIZACIÓN

Para usar el software del control de este Amplificador de Audio se debe seguir ciertos pasos, en caso que el usuario coma algún error el software informará al usuario cual fue error que se cometió.

Se comienza ejecutando el programa MATLAB, versión 6.0 (R12) o superior[18].

Después de haber ejecutado el software MATLAB, se procede a seleccionar la carpeta donde se encuentra el programa; es decir , donde se encuentran los archivos *.m que corresponden al software del amplificador de potencia controlado por computadora. Esto se lo hace en la parte de la pantalla que indica *CURRENT DIRECTORY.*

Luego se debe escribir la palabra *inicio* en la parte de la pantalla correspondiente al *COMMAND WINDOW*, a continuación se muestra la pantalla de MATLAB (figura 3.48).

---

[18] En caso de correr el programa con MATLAb versión 5.3 (R11) o una versión más antigua el programa se ejecutará pero al llegar a la pantalla para seleccionar la tarjeta de adquisición de datos a utilizar para las entradas analógicas o para las entradas / salidas digitales no reconocerá la tarjeta PCI-DAS 1000.

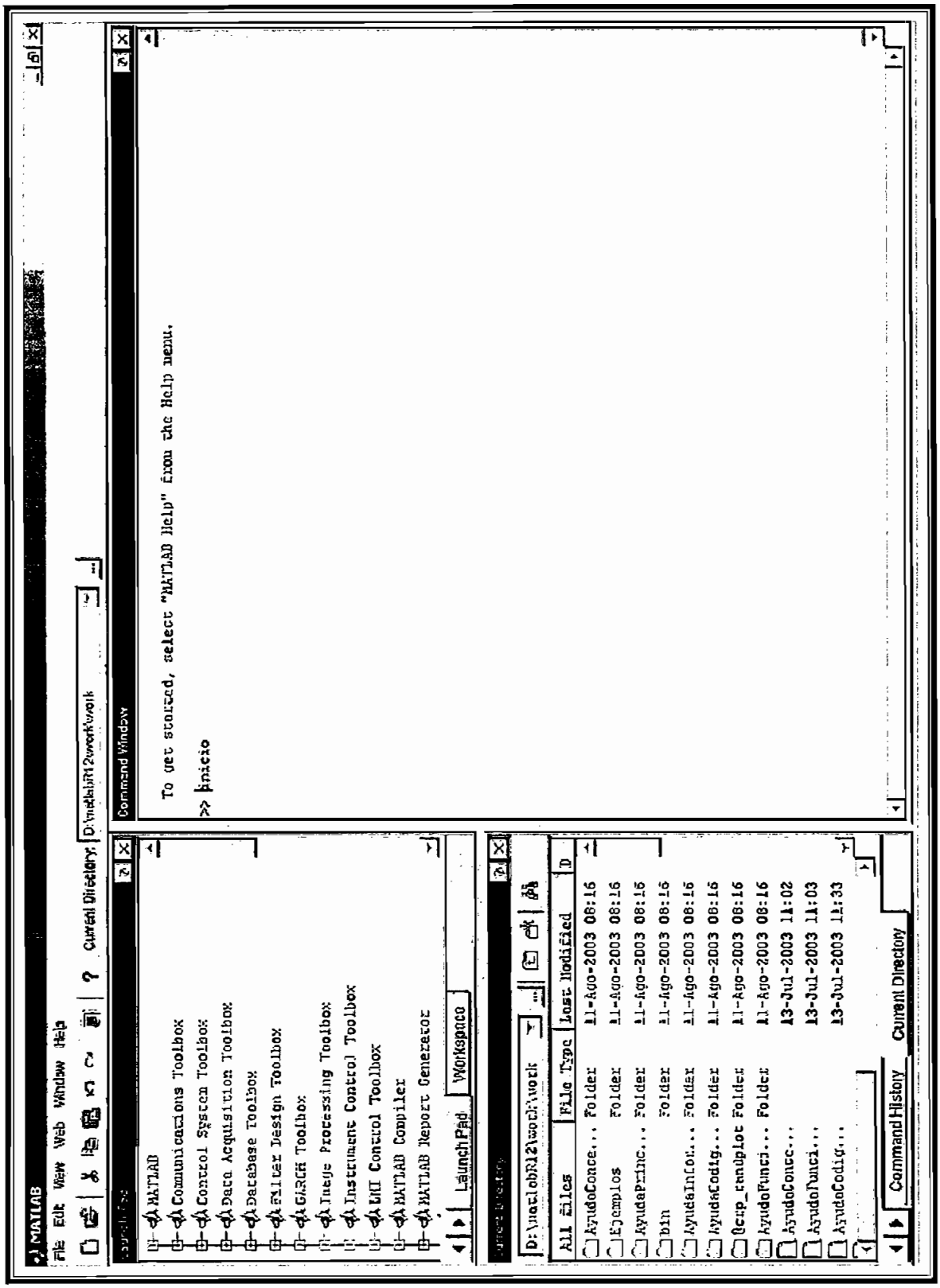


FIGURA 3.48) Programa Matlab

Después de ejecutar la línea *Inicio* se desplegará la pantalla de bienvenida durante un cierto tiempo, el cual se indicó en el numeral 3.6 (figura 3.38), luego se muestra la pantalla de configuración de los subsistemas (entradas analógicas, salidas analógicas, entradas / salidas analógicas) en la cual se debe seleccionar los subsistemas a utilizar para el control del amplificador de audio controlado por computadora. En caso de seleccionar una tarjeta que no tenga las características necesarias que se requieren para el control el software no permitirá continuar, indicando al usuario el mensaje correspondiente al error cometido.

Después de seleccionar las tarjetas que permitan realizar la adquisición de datos el software procederá a llamar al programa *controlador*, el cual se encuentra almacenado en varios archivos *.m, donde el programa principal es *controlador.m*, mencionado programa permite realizar el control del amplificador de audio; es decir, manipular las entradas analógicas (micrófono 1 y 2, auxiliares 1 y 2), además manipular la potencia de salida del amplificador, así como también poder saber cual es la potencia que está entregando el amplificador, además se puede saber cual es la carga conectada a la salida del amplificador; es decir, cual es el valor de resistencia del parlante que se encuentra conectado a la salida del amplificador.

También se cuenta con un archivo el cual contiene ayuda para el manejo del software.

CAPÍTULO 4:
PRUEBAS EN CONJUNTO

4.1 INTRODUCCIÓN

El presente capítulo tiene por objeto mostrar el trabajo en conjunto del sistema (Amplificador de Potencia de Audio Controlado por Computadora); es decir, el trabajo entre el amplificador de potencia de audio, los preamplificadores y del computador con la tarjeta de adquisición de datos.

4.2 PARTES DEL EQUIPO

El equipo construido consta de las siguientes partes partes :

- –El equipo amplificador de audio
- –Bus de datos
- –Tarjeta de adquisición de datos
- –Software de manejo

El equipo amplificador de audio está dentro de una caja de 41 cm de largo, 41 cm de ancho y 9 cm de alto, está compuesta por una etapa preamplificadora, una etapa de potencia, una etapa de protección, una etapa de medición de carga y la fuente de alimentación, la disposición de estas etapas está dada como se muestra en la figura 4.1.

El bus de datos es del tipo SCSI sirve para la interconexión entre el equipo amplificador y el computador.

La tarjeta de adquisición de datos se la utiliza como interfaz entre el equipo amplificador y el computador, su utilización ya se la indicó anteriormente.

El software se lo utiliza para el manejo del equipo amplificador de audio, su utilización y funcionamiento ya se lo vio anteriormente.

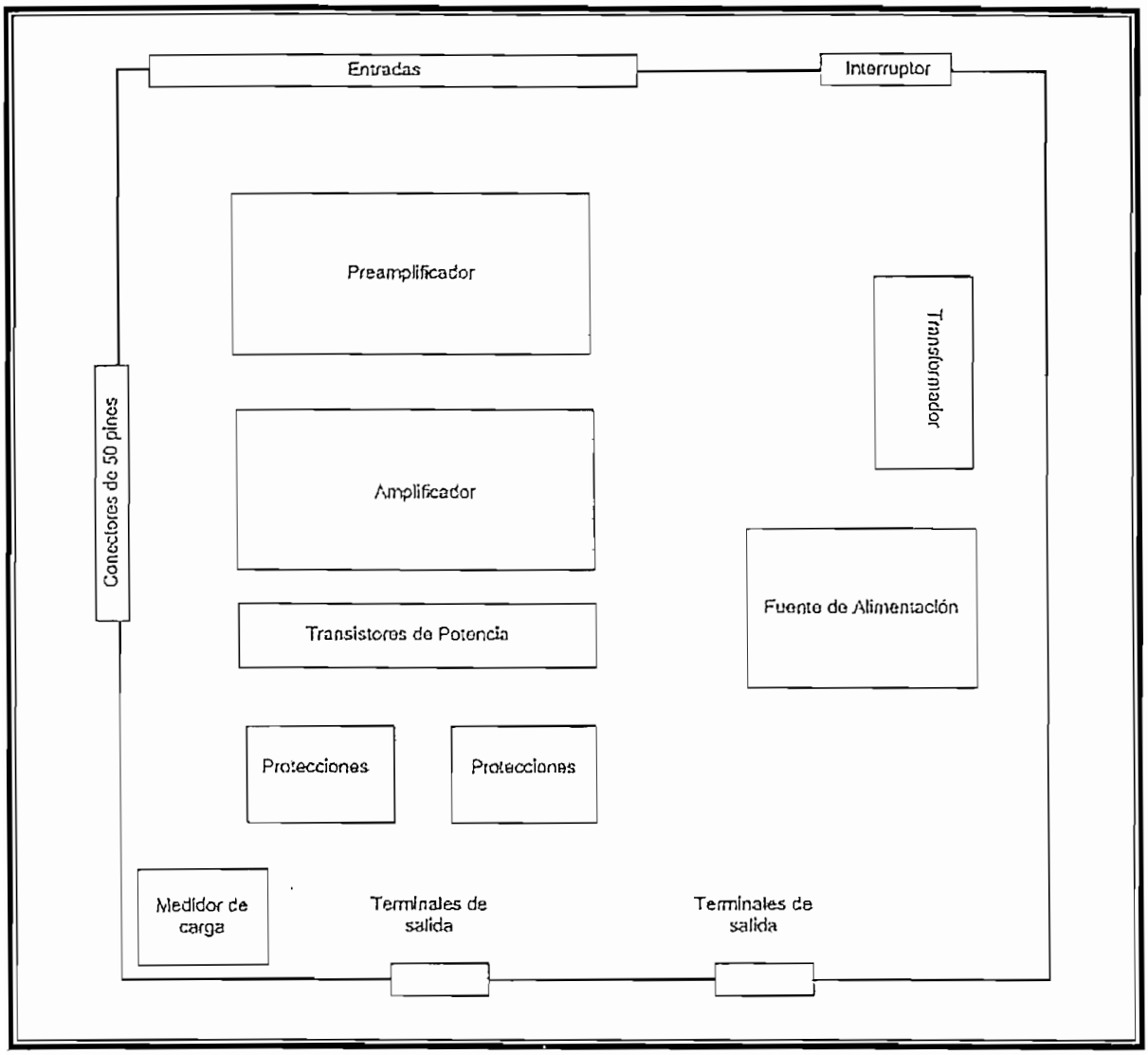


FIGURA 4.1) Disposición interna de las etapas en el equipo amplificador de audio

4.3 FUNCIONAMIENTO Y MODO DE OPERACION

El equipo construido es un amplificador de audio estereo capaz de entregar 25 vatios por canal, indistintamente de la carga (4, 6 u 8 ohmios) que se conecte a sus salidas.

Este es un equipo que puede trabajar mediante el control del computador (con lo que se tiene algunas ventajas) o sin el control del computador (ciertas restricciones).

Para operar el equipo mediante el computador colocamos el bus de datos en la parte de los conectores de 50 pines. Se enciende el equipo y mediante el software se opera el amplificador de audio, con las caracteristicas antes citadas.

Para manejar el equipo amplificador, sin la utilización del computador, se enciende el equipo y se ingresa la señal a amplificarse por los terminales RCA, (estan junto a los conectores de 50 pines). La restricción que se presenta es que se tiene una ganancia de potencia fija.

La operación detallada de cada una de partes que conforma el equipo amplificador, así como el manejo del software de utilización ya se lo indicó anteriormente.

4.4 PRUEBAS

Primeramente se calibra el manejo de los potenciometros digitales en conjunto con el software según los datos dados en el capítulo 1 (sensibilidad de entrada del amplificador).

Adicionalmente se escucha si existe algun tipo de distorsión en la reproducción del sonido, si existe se procede a manipular la parte de ecualización (incluida en el software).

La principal prueba para comprobar el buen funcionamiento del sistema fue introducir una señal de 1 KHz (usando un generador de señal) por cada una de las entradas (Micrófono o Auxiliares) y con la manipulación de los potenciómetros mediante el interfaz de usuario se mide la potencia de salida. La potencia de salida se la midió de dos formas: la primera midiendo el voltaje de salida y aplicando la fórmula $P = V^2 / R$, esto es considerando que la red de zobel elimina el efecto inductivo de la bobina del parlante; mientras que la segunda forma fue midir el voltaje de salida y la corriente que ingresa al parlante usando dos multímetros.

Al medir los voltajes de salida y de entrada al amplificador se sabe cual es el valor de la ganacia de voltaje de dicho amplificador, este valor se usa para conocer cual es el valor máximo de voltaje que debe ingresar al amplificador para tener un máximo de 25 vatios a la salida (esta manipulación se la hace mediante software).

Se comparó la señal que muestra el sofware con la señal recibida por el osciloscopio para comprobar que el software entrega una señal semejante a la señal que la tarjeta de adquisición de datos.

Mediante el osciloscopio y con ayuda del software se comprueba que se esté funcionando correctamente el mezclador (suma de las señales de entradas).

Con ayuda del generador de señal se introdujo una señal de 1 KHz la cual se la filtró mostrándose los resultados tanto en el osciloscopio como en el software.

4.5 RESULTADOS

Los resultados obtenidos del equipo en conjunto están apegados a los resultados parciales de cada una de las etapas antes vistas. Sin embargo se nota una pequeña distorsión en la señal de salida pero se la puede minimizar con el uso de los filtros (ecualizador).

Además existe un retardo mínimo entre la señal de entrada y la señal de salida del equipo amplificador, esta deficiencia es inebitable porque los datos que se obtienen de la tarjeta de adquisición de datos hay que procesarlos y volverlos a enviar al exterior por medio de la tarjeta de adquisición de datos.

4.6 EQUIPOS DE MEDICIÓN

Los equipos utilizados en la toma de medidas se muestran a continuación:

EQUIPOS	Marca	Modelo	Características
GENERADOR DE SEÑAL	YAMATSU	AG200CD	Audio Generator / Counter
OSCILOSCOPIO	HAMEG	HM 605	60 MHz Oscilloscope
MULTÍMETRO	FLUKE	88	Automotive Meter

TABLA 4.1) Equipos utilizados en las mediciones

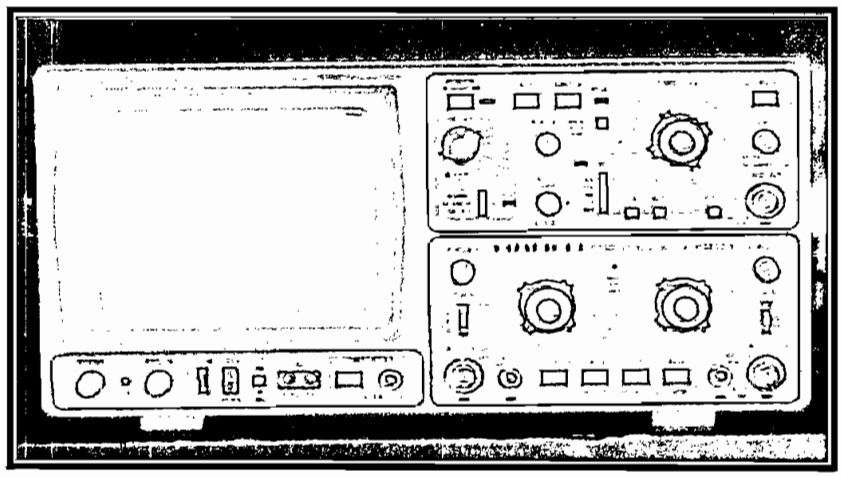


FIGURA 4.2) Osciloscopio

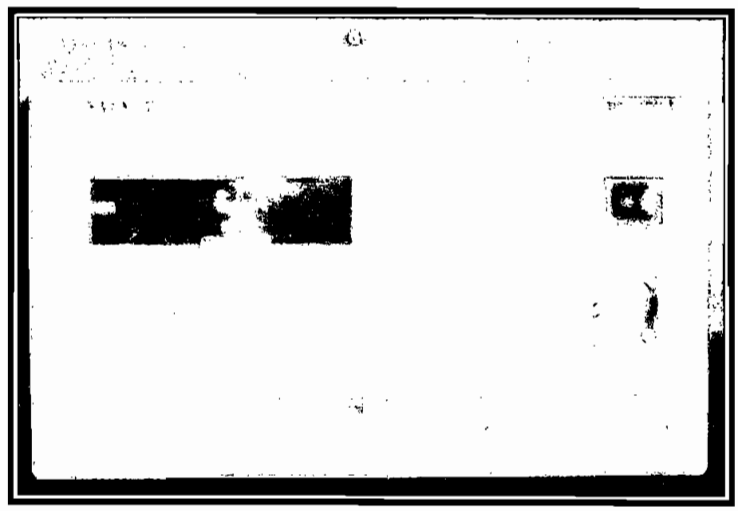


FIGURA 4.3) Generador de señal

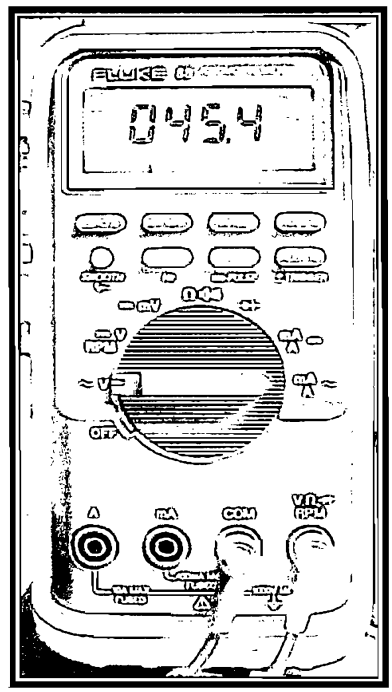


FIGURA 4.4) Multimetro digital

4.7 VISIÓN GLOBAL DE SISTEMA

A continuación se presenta una fotografía en la que se muestra como está conformado internamente el equipo amplificador de audio:

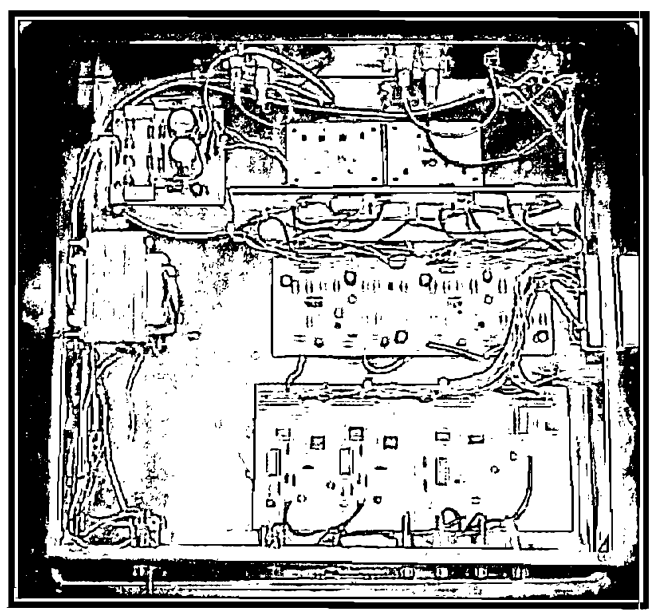


FIGURA 4.5) Estructura interna del equipo amplificador

Un sistema global (computador, equipo amplificador, parlantes) se muestra a continuación:

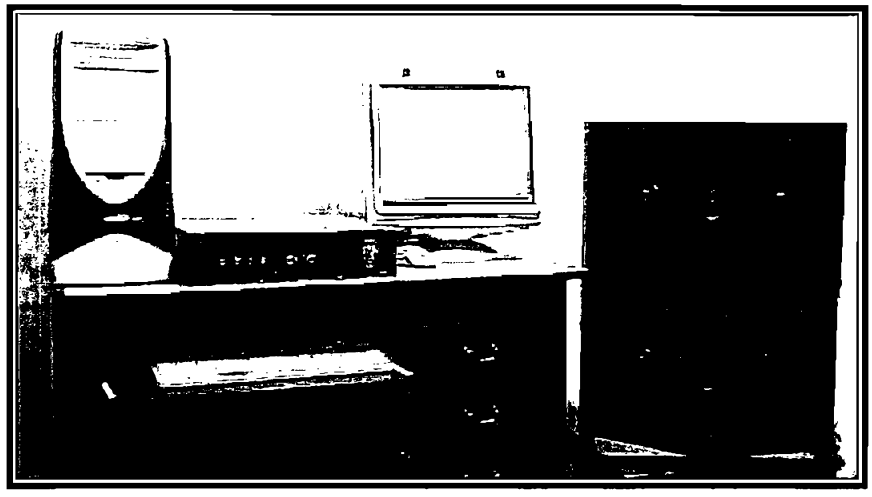


FIGURA 4.6) Sistema global

4.8 COSTO DEL PROTOTIPO

A continuación se detalla el costo del sistema, primero se indica el costo referente a sus diferentes partes y al final se indica el costo global del sistema.

Fuente de Poder (27 Voltios DC)		
Cantidad	**Elemento**	**Costo (USD)**
1	Placa	5
1	Transformador (toma central) 110/20 V	20
2	Disipadores	1
4	Diodos rectificadores 6 A	2
2	Capacitores 4700 uF/50V	4
2	Resistencias de potencia	1
1	7818	1
1	7819	1
3	Fusibles y portafusibles	1.5
2	Capacitores 10 uF/50V	0.5
	Otros	2
TOTAL		**39**

TABLA 4.2) Costo de los elementos que conforman la fuente de poder

Etapa de Preamplificacion (4 entradas)		
Cantidad	Elemento	Costo (USD)
1	Placa	20
4	TL084 con socalos	4
7	Potenciometros digitales (Xicor) con socalos	42
1	Conector para microfono estereo	1
1	Conector para microfono monofonico	1
4	Conectores RCA	2
	Resistencias y capacitores	3
	Otros	2
TOTAL		75

TABLA 4.3) Costo de los elementos que conforman la etapa de preamplificación

Etapa de Amplificacion (25 Vatios)		
Cantidad	Elementos	Costos (USD)
1	Placa	20
1	Disipador	3
4	ECG130	8
4	ECG290A	2
4	ECG130	8
2	ECG291	2
2	ECG292	2
4	Resistencias de potencia	1.5
	Capacitores, resistencias, diodos	4
	Otros	4
TOTAL		54.5

TABLA 4.4) Costo de los elementos que conforman el amplificador

Etapa de Proteccion		
Cantidad	Elementos	Costos(USD)
2	Placas	10
4	Diodos 1N4007	1
2	ECG	1
2	ECG	1
	Resistencias	1
	Otros	2
TOTAL		16

TABLA 4.5) Costo de los elementos que conforman la etapa de protección

Interfaz Hardware		
Cantidad	Elemento	Costo (USD)
1	Tarjeta de Adquisision de Datos (con extras)	600
	Envio (MiddleBoro-Quito)	200
2	Conectores de 50 pines	10
4	Borneras	6
	Otros	10
	TOTAL	826

TABLA 4.6) Costo de los elementos que conforman el interfaz hardware

Etapa de medicion de Impedancia		
Cantidad	Elementos	Costo (USD)
1	Placa	5
1	Interruptor	0.35
	Resistencia	1
	Otros	2
	TOTAL	8.35

TABLA 4.7) Costo de los elementos que conforman la etapa de medición

Presentacion del prototipo		
Cantidad	Elementos	Costo (USD)
1	Caja Metalica (pintura, ranuras, orificios)	30
	pernos, tornillos	2
	Interruptores, luz piloto	4
	Otros	10
	TOTAL	46

TABLA 4.8) Costo de accesorios de presentación del prototipo

Costo del prototipo	
Descripción	Costo (USD)
Fuente de poder	39
Etapa de preamplificación	75
Etapa de amplificación	54.5
Etapa de protección	16
Interfaz hardware	826
Etapa de medición de impedancia	8.35
Presentación del prototipo	46
Mano de obra	500
TOTAL	1564.85

TABLA 4.9) Costo del sistema de amplifiación de audio controlado por computadora

CAPÍTULO 5:
CONCLUSIONES Y RECOMENDACIONES

5.1 CONCLUSIONES

- Los resultados obtenidos experimentalmente, están de acuerdo con los valores que se encontraron teóricamente, las diferencias se marcan por las características propias de los elementos discretos que se utilizan.

- Actualmente en el mercado se puede encontrar diferentes formas de realizar la programación para cumplir con el objetivo de controlar un amplificador de audio por computadora, todos tienen sus características (en el sentido de dificultad de programación) que le diferencian de los demás así como también el costo, como por ejemplo: existe el programa SoftWire (herramienta para Visual Basic), el cual permite programar gráficamente la tarjeta de adquisición de datos con Visual Basic, con este software es muy sencillo realizar la programación pero el problema principal con este software es el costo (alrededor de 400 dólares). Otro Programa para realizar la programación de las tarjetas de Adquisición de Datos es LabView, del cual no se tenia mucho conocimiento, por estas y otras razones más se realizó la programación con Matlab, ya que se tenía el suficiente conocimiento del funcionamiento, bondades y ventajas de este software, ya que se conocía como realizar interfaces gráficas, manipulación de información y otras operaciones importantes.

- Un parlante es un transductor cuya impedancia tiene una componente inductiva por estar conformado por una bobina, es decir no es totalmente resistiva, por lo que es necesario que un amplificador de potencia de audio tenga presente a la salida una red de Zobel para eliminar el efecto de la componente inductiva del parlante (bobina), de tal manera que no exista mas demanda de energía por parte del parlante para altas frecuencias.

- El amplificador de potencia de audio debe contar con protecciones frente a cortocircuitos a la salida ya que si se conecta un parlante, el cual puede estar cortocircuitado, no se tiene conocimiento del correcto

funcionamiento puede dañar al amplificador, así que para cualquier diseño se debe prever algún tipo de protección frente a este tipo de acontecimientos.

- La construcción de cualquier sistema electrónico es muy complicada y costosa, ya que no siempre se puede encontrar en el mercado local ciertos dispositivos electrónicos o elementos pasivos con los cuales se puede hacer una excelente construcción, como por ejemplo potenciómetros digitales es muy complicado conseguir, los potenciómetros digitales utilizados en este proyecto fueron importados de Alemania, la tarjeta de adquisición de datos fue importada de Estados Unidos, la importación de estos elementos aumenta el valor original a casi el doble de su precio, conectores de 50 pines no se encontró en el mercado; es decir, no se tiene las facilidades necesarias para poder construir un cierto sistema electrónico a bajo costo.

- Actualmente se puede comprar tarjetas de Adquisición de datos que se las puede conectar al puerto USB o también tarjetas para el slot PCMCIA para las Laptop's, con estas tarjetas se puede tener una aplicación portátil; es decir, para aplicaciones del amplificador de audio en sonido en vivo como lo son eventos en distintos lugares, se requiere que el computador que realice el control del amplificador de audio sea portátil, mientras que para aplicaciones del amplificador de audio como en teatros; es decir, en lugares fijos se puede utilizar un computador personal para realizar dicho control.

- La búsqueda y elección de los C.I. debe ser minuciosa y objetiva de acuerdo a la aplicación que se vaya a tener, analizar detalladamente algunas alternativas es importante, ya que varios C.I. pueden cumplir la misma función pero pueden tener características que se oriente al uso en cierta aplicación, por ejemplo en audio, en video, etc. Un claro ejemplo se puede observar entre el LM324 y el TL084.

- Para el caso del amplificador de potencia se tienen dos canales estructuralmente iguales, con los mismos elementos sin embargo la operación de estos no es igual, esto se debe a la diferencia que existe entre elementos de un mismo valor o un mismo tipo, por razones de construcción, por eso los fabricantes dan un margen de tolerancia de los valores de los elementos.

- La utilización de circuitos integrados es la mejor opción por las ventajas que prestan, sin embargo en mucho casos no es la mas conveniente, en nuestro caso se buscó un circuito integrado con el cual se pueda realizar el trabajo que se requería (amplificador de audio de 25 vatios), se encontró un C.I. que podía realizar el trabajo, el problema es que la información que acompaña a este C.I. es muy poca, lo que impide realizar un diseño personalizado que tenga un buen funcionamiento.

- Una recomendación que es bueno recalcar. En la construcción de un sistema con varias etapas o partes es indispensable probar y tomar medidas de cada parte independientemente, de tal manera que al realizar las pruebas en conjunto se tenga la seguridad de que una etapa no va a averiar a la otra.

- La tendencia que se tiene a controlar sistemas por computador cada día aumenta, por las facilidades que esto presenta. En forma global los métodos que se sigue para conseguir esto son similares, es decir en lugar de un amplificador de audio puede ser un sistema de válvulas, un sistema de monitoreo, etc. Lo que se trata de mostrar con este proyecto es la funcionabilidad, operabilidad y precisión que se puede tener en un sistema controlado por computador.

- En el mercado existen amplificadores de audio controlados por computador pero su interfaz de control es diferente, en si no se quiso construir algo que ya existe en el mercado, si no mas bien dar una visión

de sistemas controlados por computador y en este caso se escogió como sistema a un amplificador de audio.

- El sistema de audio estereo construido es capaz de funcionar independientemente de la computadora, con la restricción de tener una ganancia fija de aproximadamente 36.9dB y la capacidad de manejar una sola entrada estereo.

5.2 RECOMENDACIONES

- Se recomienda que la velocidad de muestreo por cada canal y la cantidad de bits de cuantización de la tarjeta de adquisición de datos debe ser muy alta para poder tener una mayor cantidad de muestras para la adquisición y envío de datos y de esta forma tener una mayor nitidez de la señal. La velocidad de muestreo puede ser 44.1 KHz y la cantidad de bits para la conversión pueden ser 16 bits.

- Para realizar la compra de una tarjeta de adquisición de datos se debe tomar en consideración cual va a ser el uso; es decir, con qué subsistemas debe contar, como por ejemplo: cuántas entradas analógicas, cuántas salidas analógicas, cuántas entradas / salidas digitales, cuántos contadores / temporizadores. Mientras más completa sea la tarjeta, es decir, si cuenta con todas las características antes mencionadas la tarjeta va a ser más costosa.

- Cuando se va a adquirir una Tarjeta de Adquisición de datos también hay que tomar en consideración si esta necesita o no algún software adicional aparte de los controladores propios del hardware, por ejemplo, las tarjetas de adquisición de datos de *Computer Boards* para trabajar con Visual Basic, a más de necesitar el software *SoftWire* necesita un software adicional UL (Universal Library), cuyo costo referencial es de 50 dólares extras al costo de la tarjeta de adquisición de datos (600 dolares

en promedio) y al costo del software SoftWire (400 dolares), dados estos requerimientos la tarjeta de adquisición de datos puede trabajar con Visual Basic, pero estamos hablando de tener un presupuesto de aproximadamente 1500 dolares, que es casi el doble del costo de la tarjeta.

- Se recomienda conectar en paralelo a los condensadores de la fuente de alimentación una resistencia por la que descarguen estos en el momento de suspender la alimentación eléctrica. Para nuestro caso, sin estas resistencias vemos que al desconectar la alimentación nuestro amplificador de audio sigue trabajando por lo menos unos 20 segundos más y comienza a perder la intensidad poco a poco causando un sonido desagradable.

- El área de contacto entre el dispositivo y el disipador es de extrema importancia para minimizar la resistencia térmica entre la carcaza y el disipador. Las superficies deben ser planas, lisas y sin suciedad, corrosión y óxidos superficiales; para mejorar la capacidad de transferencia de calor y minimizar la formación de óxidos y agentes corrosivos se aplican grasas de silicio.

- Los dispositivos deben montarse en forma correcta sobre el disipador de calor, a fin de conseguir la precisión exacta de montaje entre las superficies de contacto, estos procedimientos correctos de instalación normalmente son indicados por los fabricantes del dispositivo.

BIBLIOGRAFÍA

[1] Coughlin Robert; Driscoll Frederick, Amplificadores y Circuitos Integrados Digitales, Quinta edición, 1999.

[2] Savant C.T.; Roden Martín; Carpenter Gorden, Diseño Electrónico: Circuitos y Sistemas, Tercera edición, 2000.

[3] Boylestad Robert; Nashelsky Louis, Electrónica: Teoría de Circuitos, Editorial Prentice-Hall Hispanoamericana, México, Sexta edición, 1997.

[4] Marshall Leach; Kendall Hunt, Introduction to Electroacoustics and Audio Amplifier Design, Segunda edicion, 2001.

[5] Muhammad Rashid, Electrónica de Potencia: Circuitos, Dispositivos y Aplicación, Segunda Edición.

[6] Bapat Y.N., Dispositivos y Circuitos Electrónicos, Editorial McGraw-Hill Latinoamericana, Colombia, 1981.

[7] http://www.ecgproducts.com

[8] http://www.national.com

[9] http://www.shure.com

[10] http://www.xicor.com

[11] http://www.ni.com

[12] http://www.measurementcomputing.com

ANEXO A:
ESQUEMÁTICOS Y RUTEO DE LA PLACAS

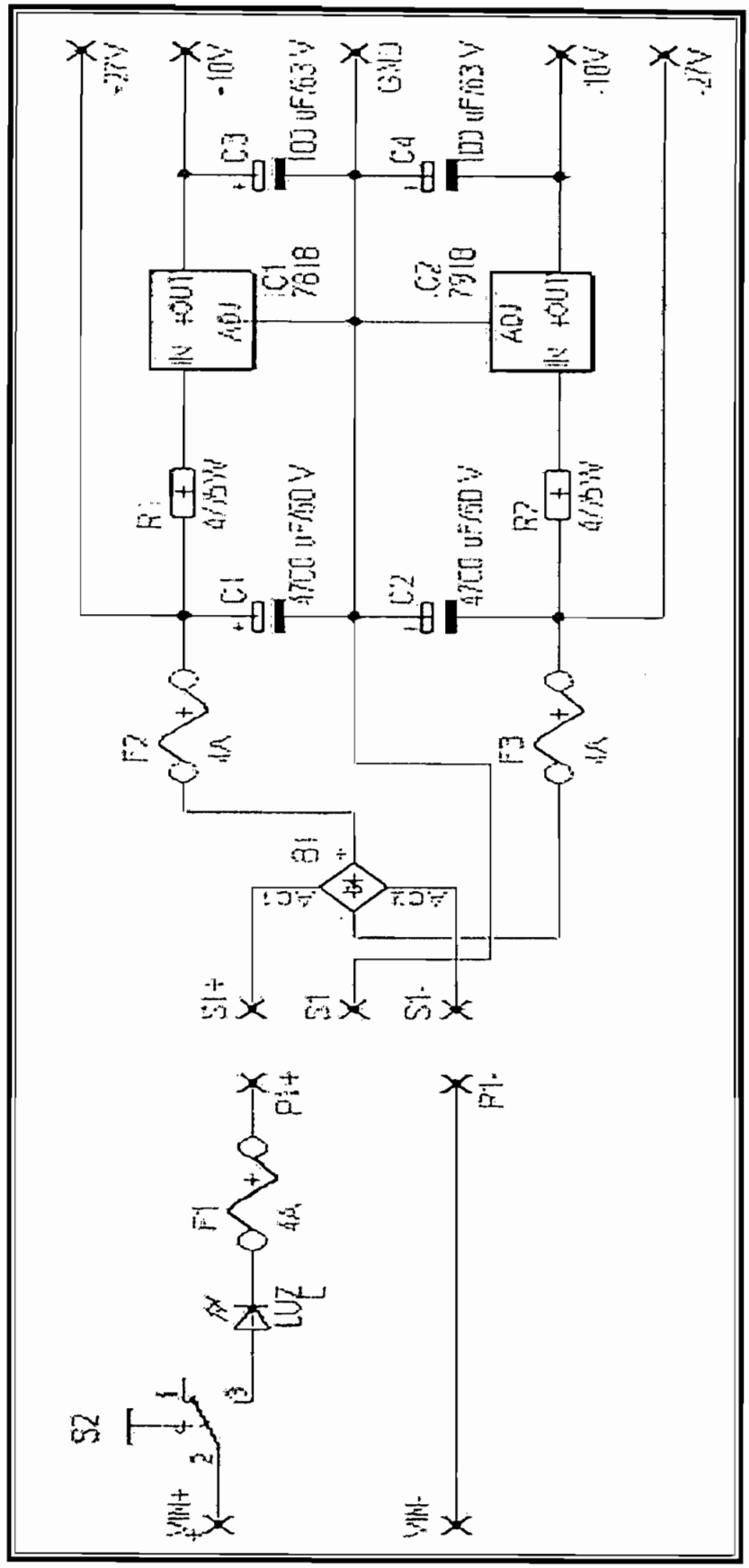


Figura A.1 Esquemático de la Fuente de Alimentación

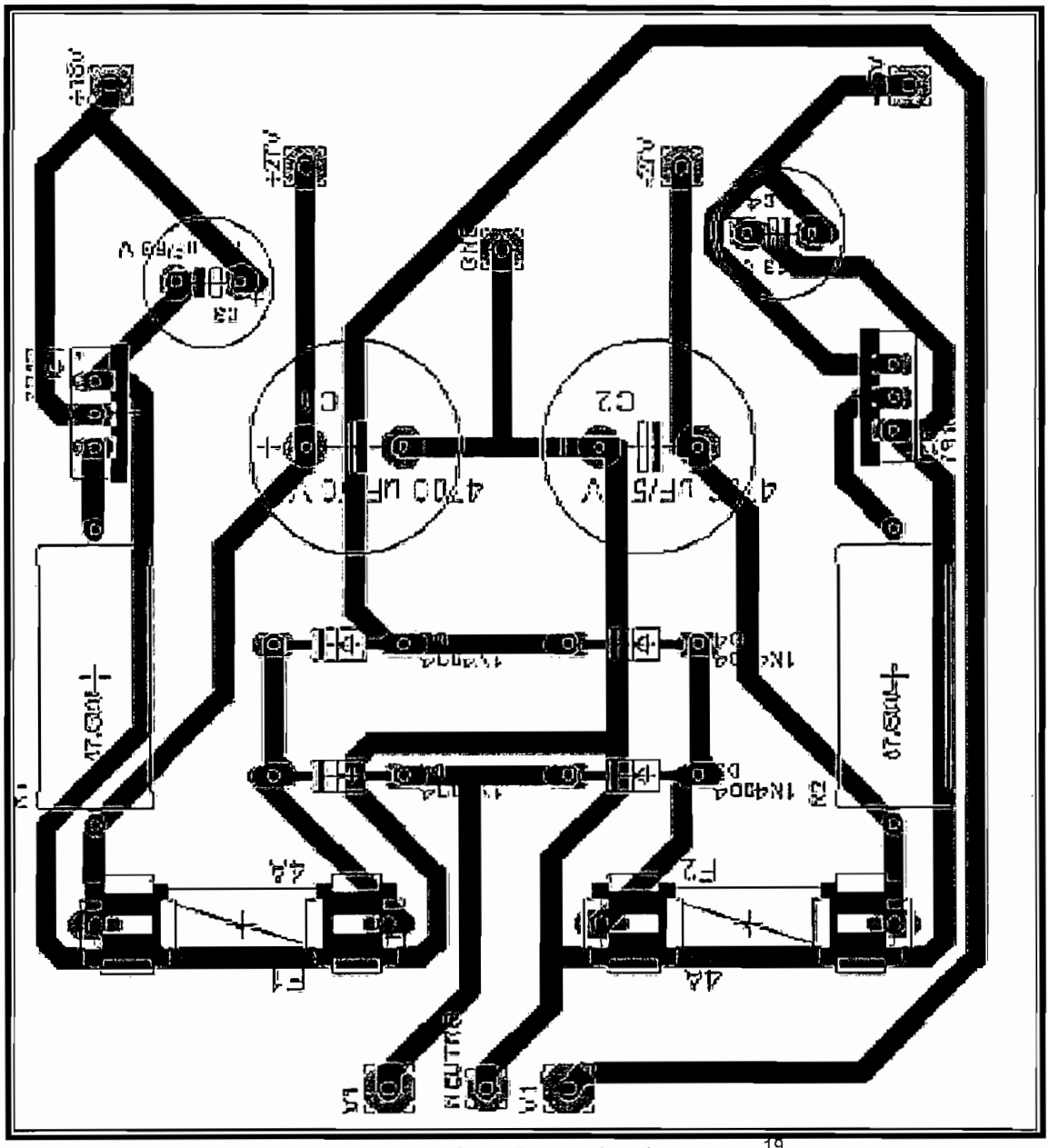


Figura A.2 Ruteo de la Fuente de Alimentación[19]

[19] En el esquemático mostrado en la figura A.2 solo se muestra la fuente a partir del secundario del transformador, no se toma en consideración el primario del transformador, ni el fusible del primario, ni tampoco del interruptor ya que estos elementos no van conectados *directamente* en la baquelita.

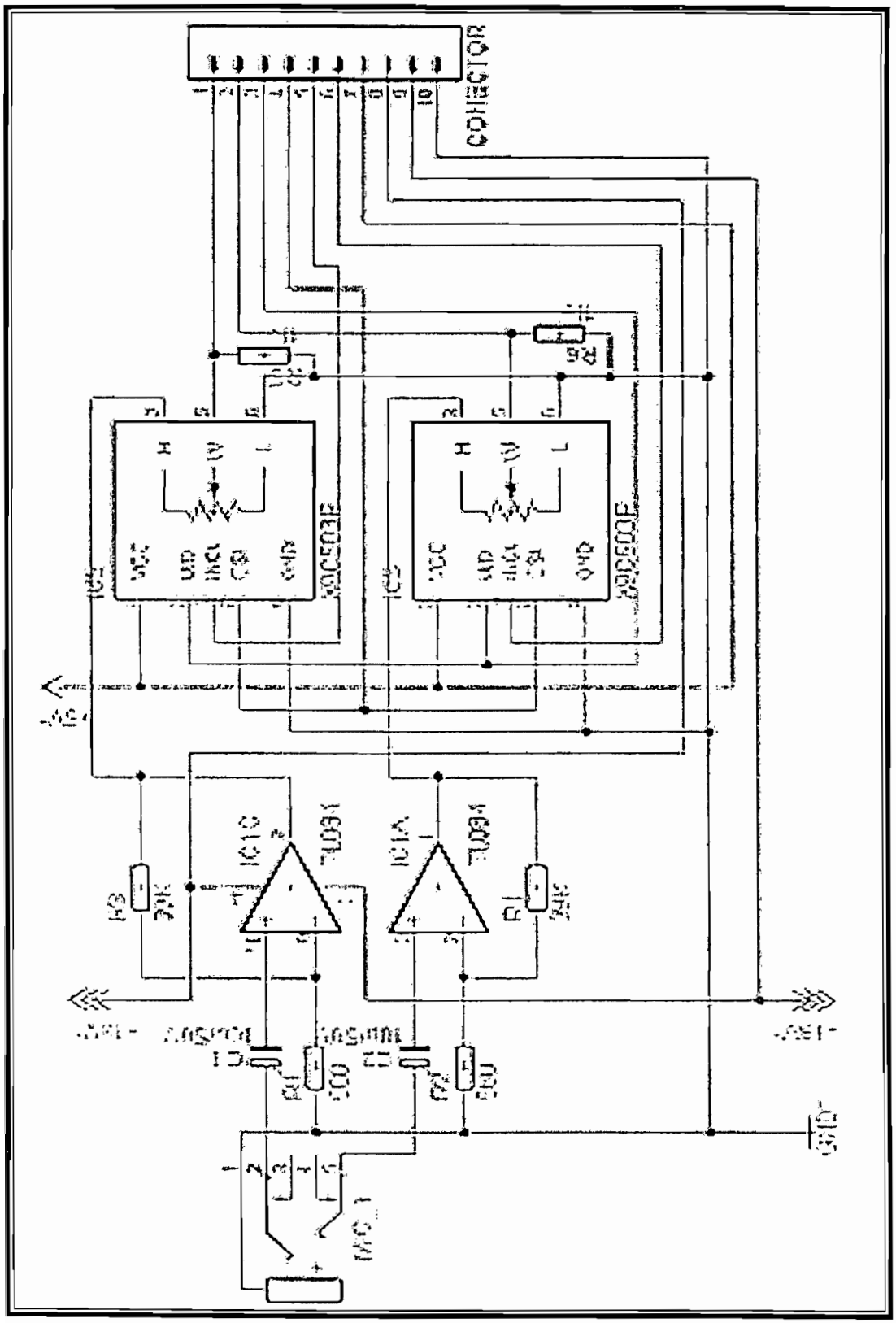


Figura A.3 Esquemático del Preamplificador de Micrófono Estereofónico

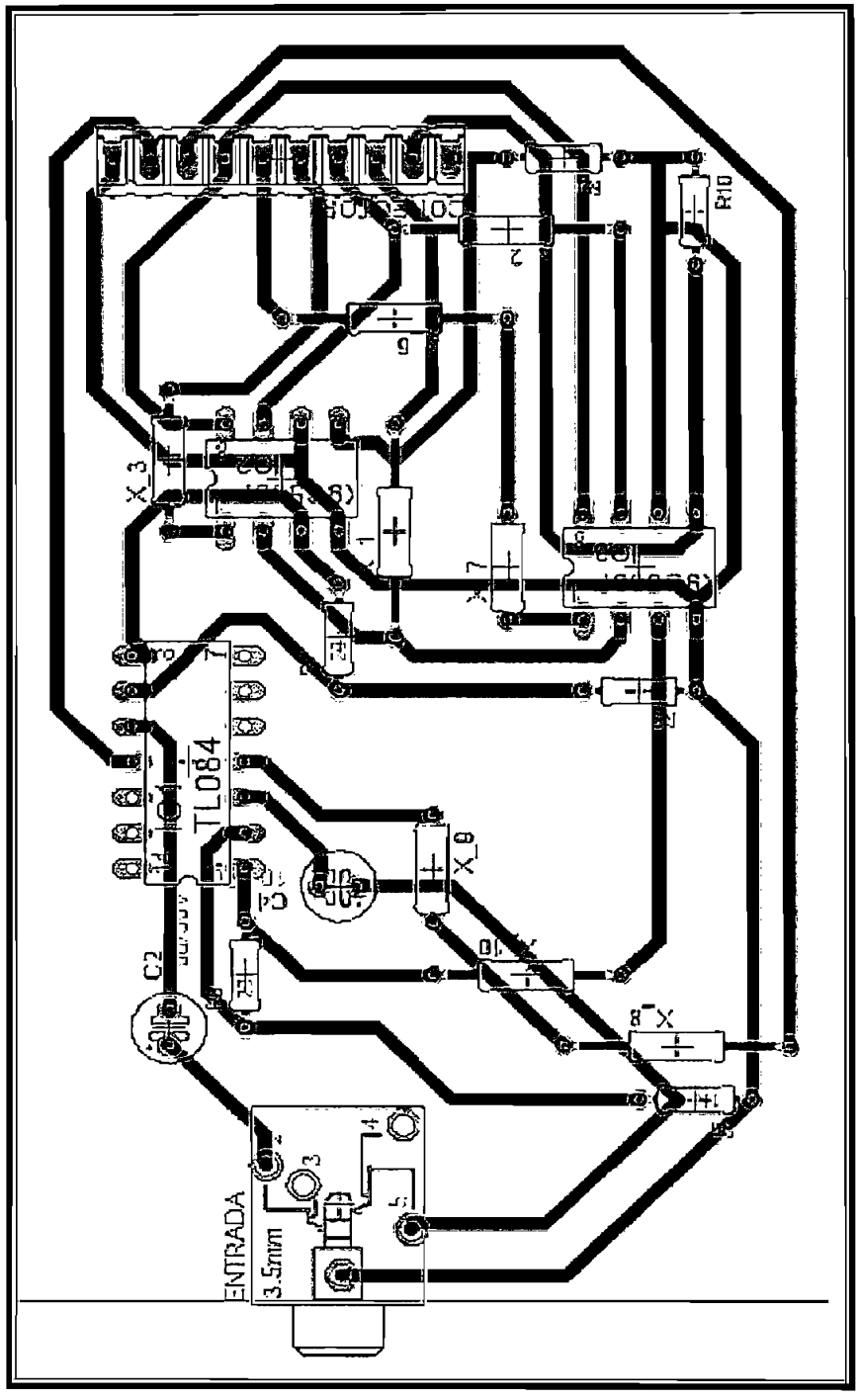


Figura A.4 Ruteo del Preamplificador de Micrófono Estereofónico[20]

[20] Cabe Indicar que en los preamplificadores los elementos que se denominan X_# son cortocircuitos que se los puso la baquelita para poder realizar el ruteo completo en un solo lado.

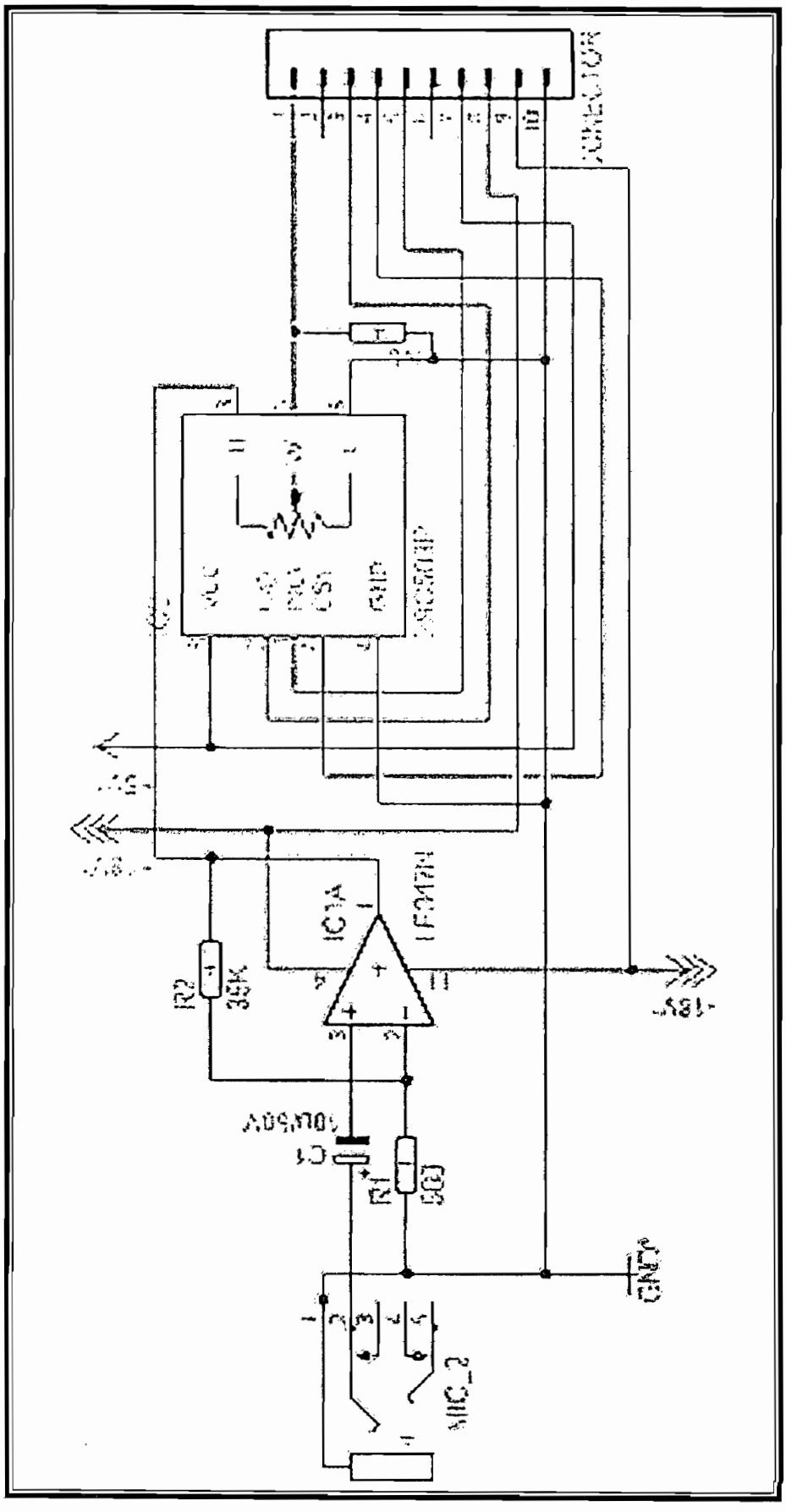


Figura A.5 Esquemático del Preamplificador de Micrófono Monofónico

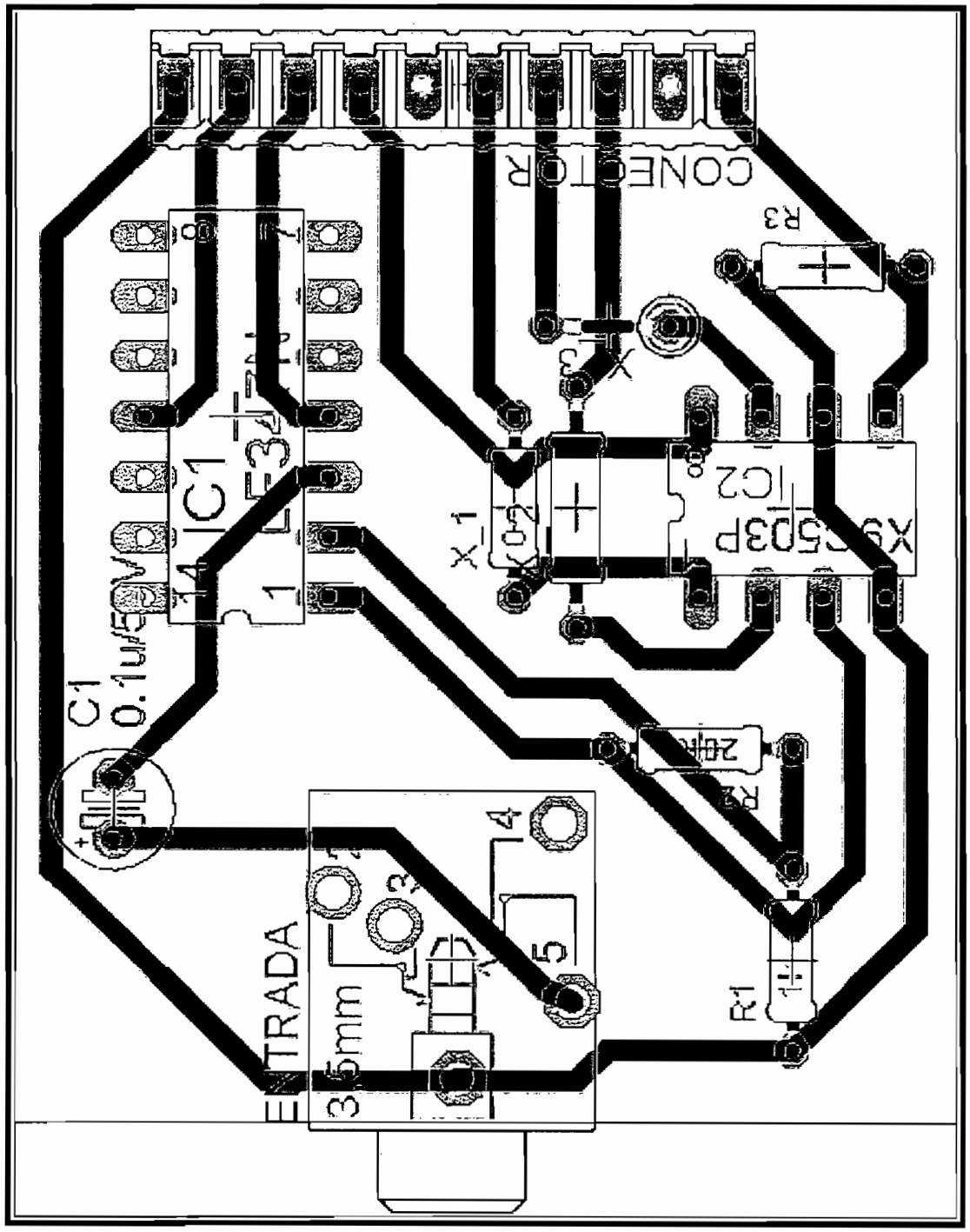


Figura A.6 Ruteo del Preamplificador de Micrófono Monofónico

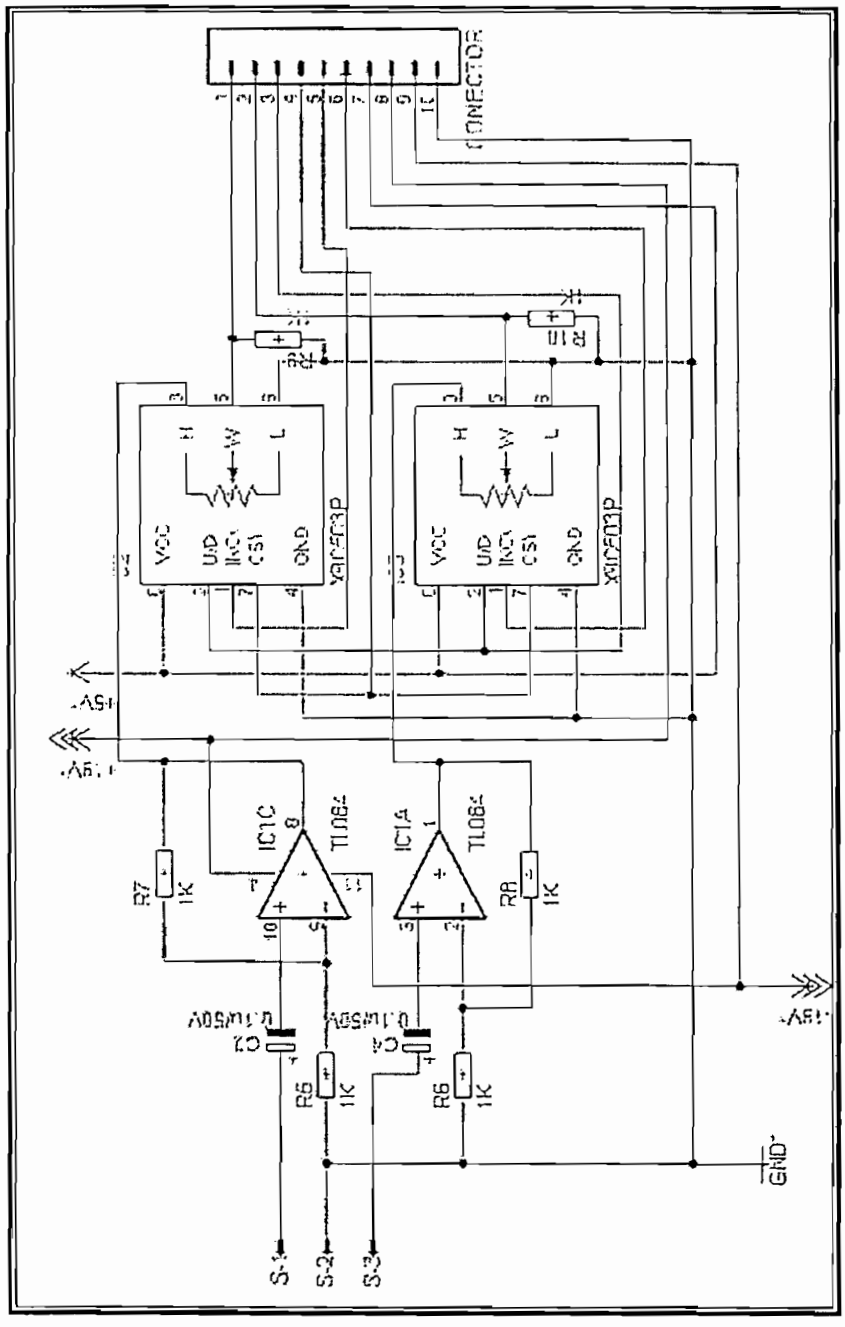


Figura A.7 Esquemático del Preamplificador de los Auxiliares Estereofónicos

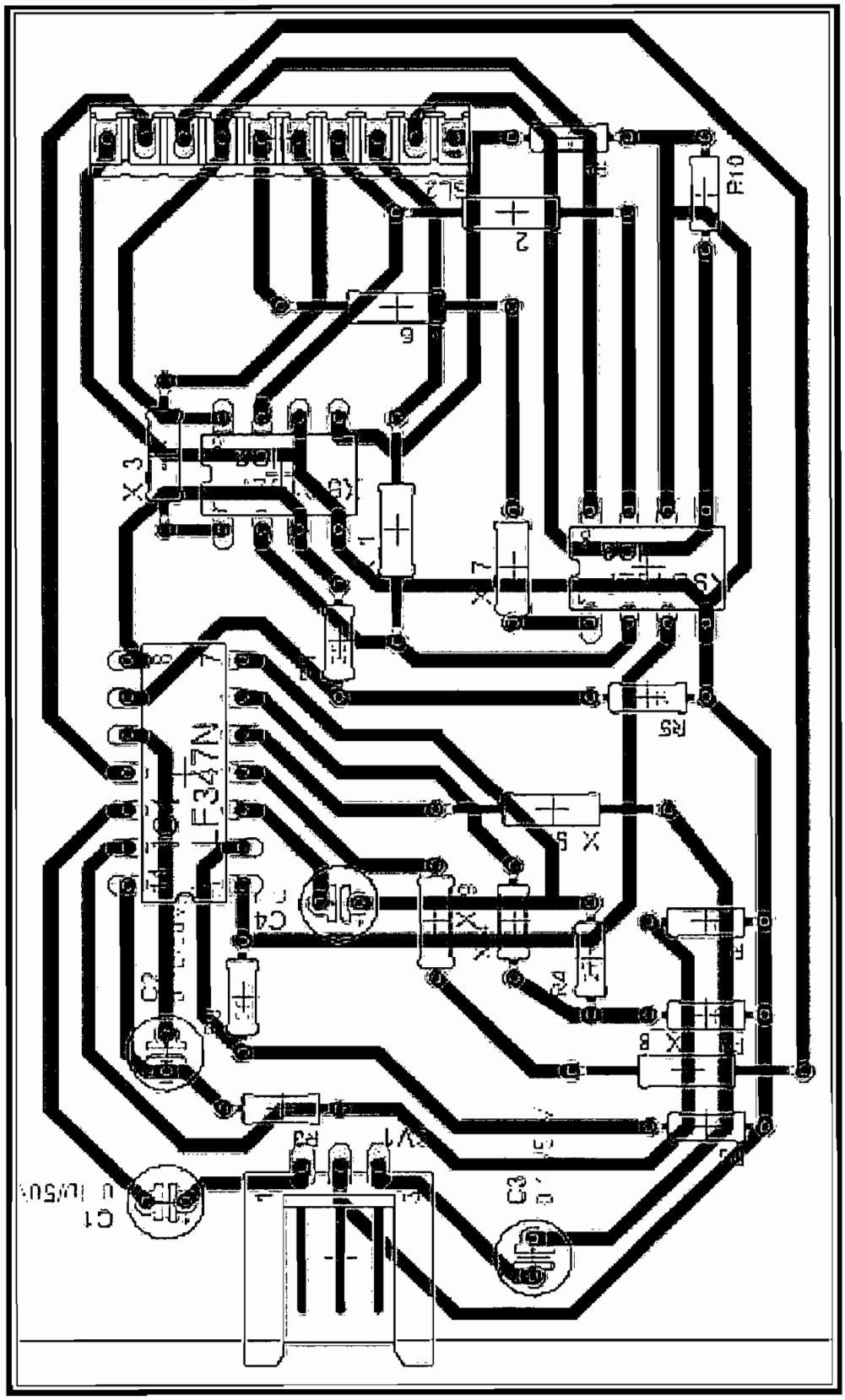


Figura A.8 Ruteo del Preamplificador de los Auxiliares Estereofónicos

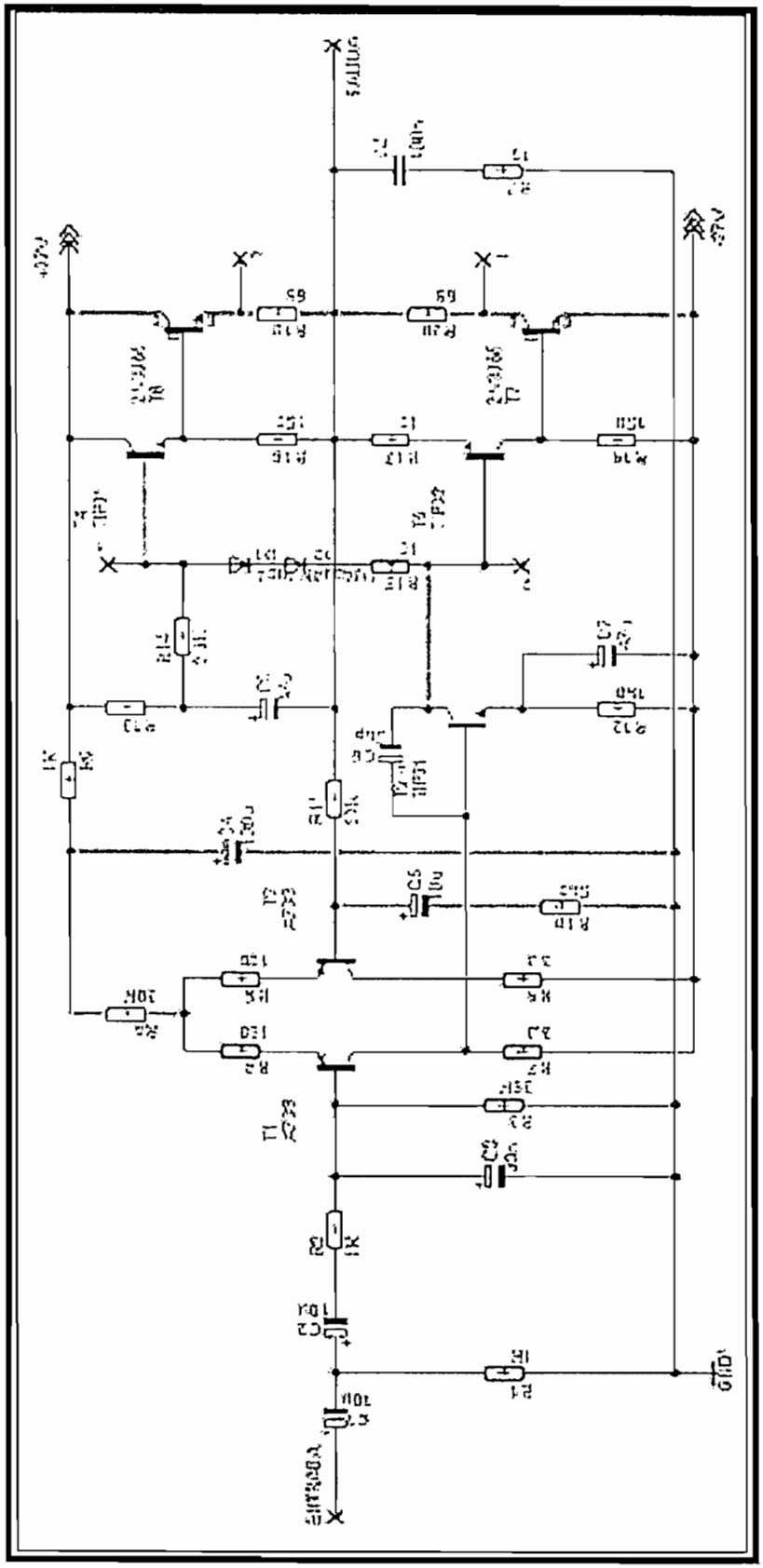


Figura A.9 Esquemático de un canal del Amplificador de Audio

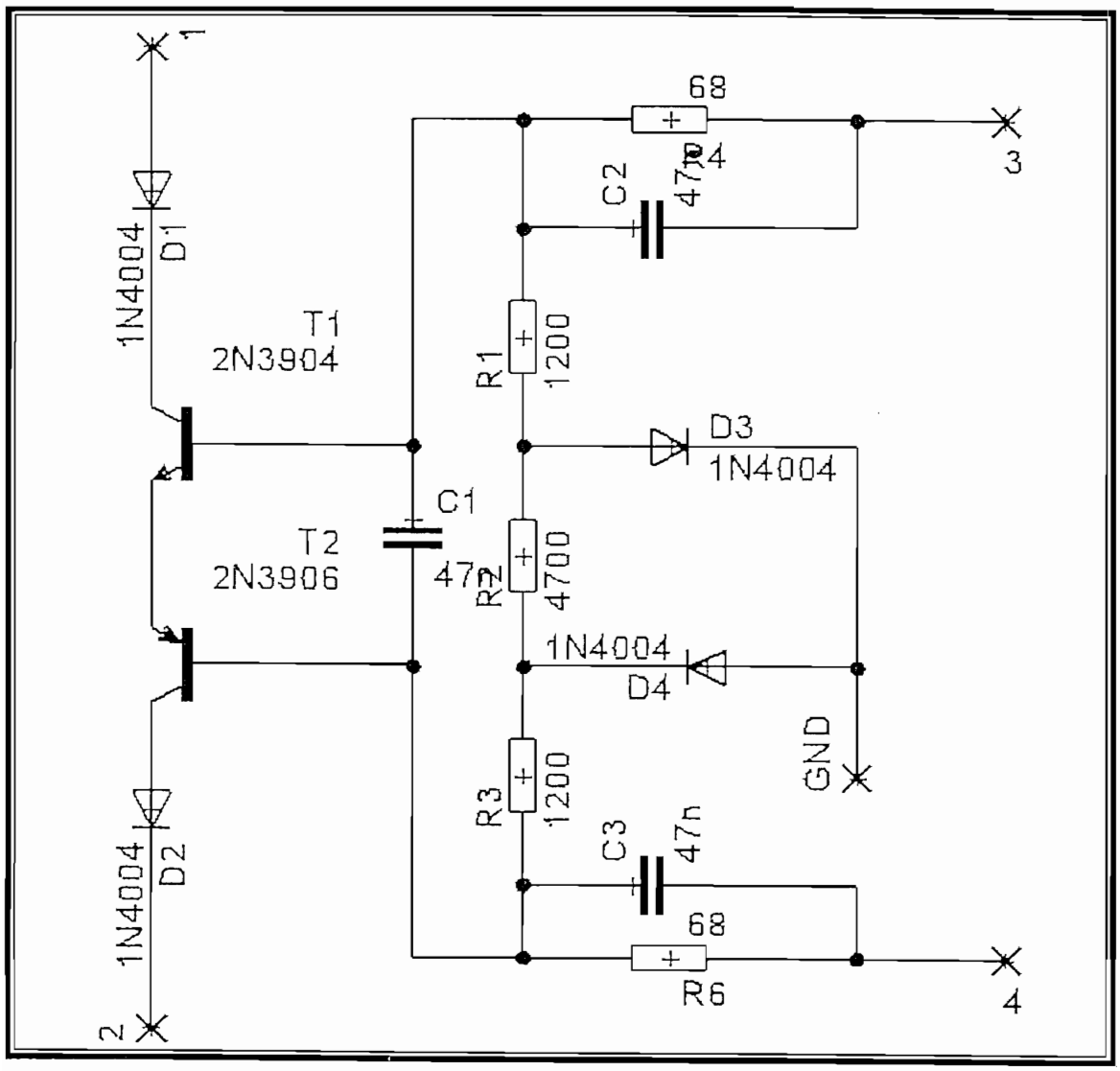


Figura A.10 Esquemático del circuito de protección

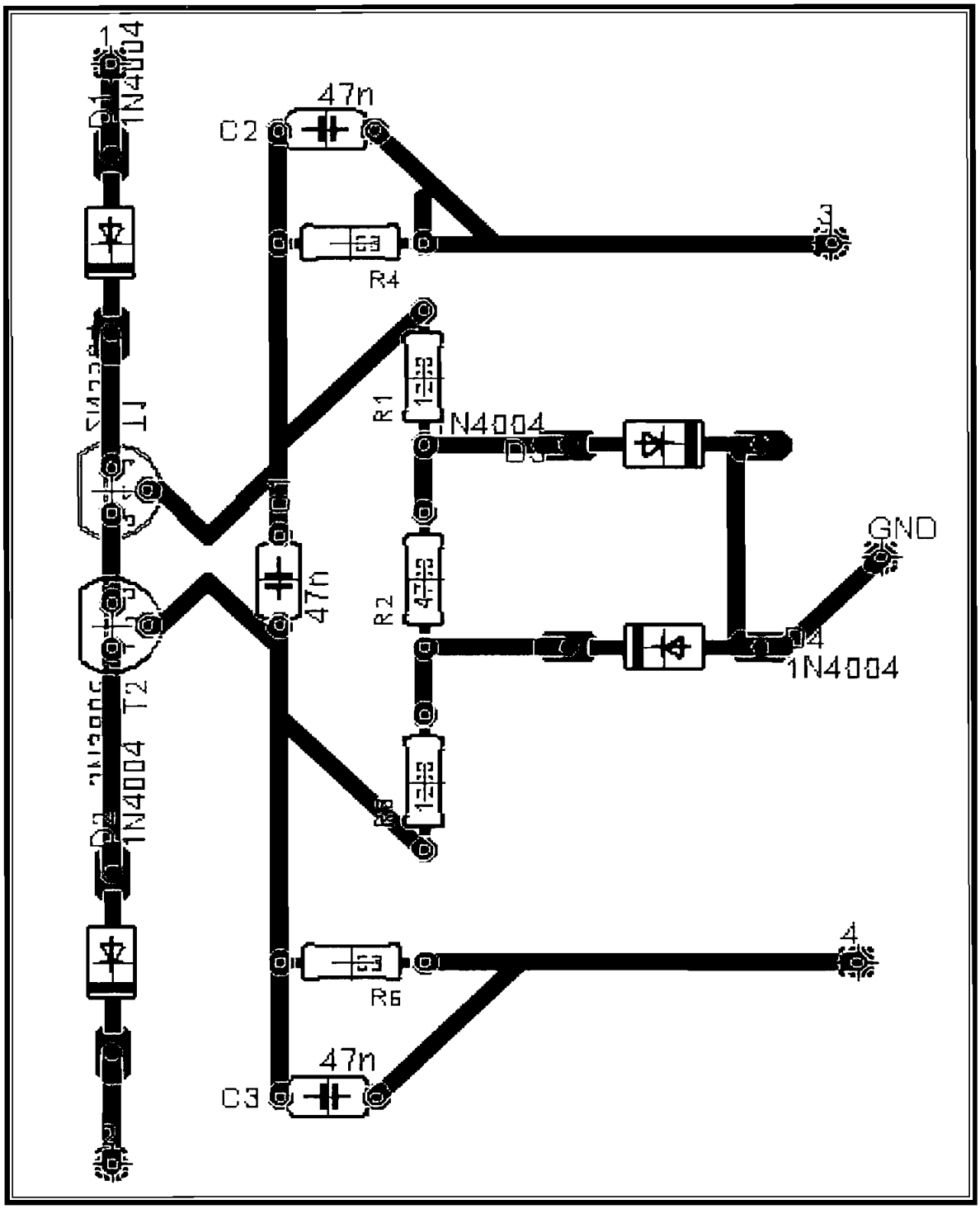


Figura A.11 Ruteo del circuito de protección

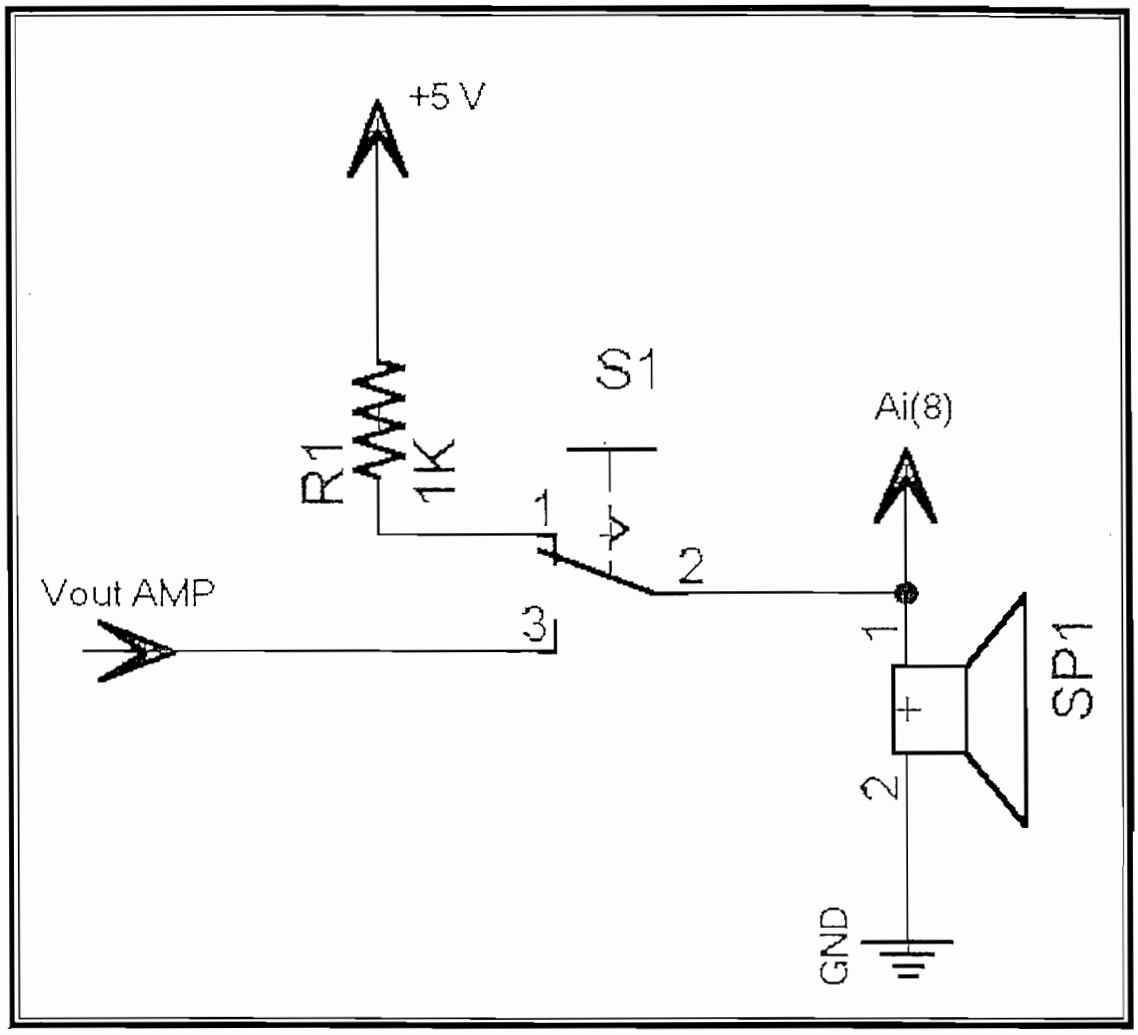


Figura A.12 Esquemático del circuito para la medición de la resistencia del parlante

ANEXO B:
CÓDIGO DE LOS PROGRAMAS

INICIO

```
% ========== Inicio ========== %
function Inicio
% Figura y Elementos ===============
uni='Normalized';
posicionfig=[0.25 0.3 0.5 0.4];
colorfondo=[0.9255 0.9137 0.8471];
colortxt=[0.3 0.3 1];
figini=figure(...
   'Units',uni,...
   'Color',colorfondo,...
   'Menubar','none',...
   'Name','Inicio',...
   'NumberTitle','off',...
   'Position',posicionfig,...
   'ToolBar','none',...
   'Resize','off',...
   'Visible','off',...
   'WindowStyle','Modal');

% Texto "Amplificador de Potencia" ===============
uicontrol(...
   'Units',uni,...
   'BackgroundColor',colorfondo,...
   'FontSize',18,...
   'FontWeight','bold',...
   'ForegroundColor','b',...
   'Horizontalalignment','center',...
   'Position',[0.1 0.6 0.8 0.35],...
   'String','AMPLIFICADOR DE POTENCIA CONTROLADO POR
COMPUTADORA',...
```

```
    'Style','text');

% Texto "Integrantes" ==============
integrantes={'Integrantes:    Alvarez Victor'
             '                Arrobo Gabriel'};
uicontrol(...
    'Units',uni,...
    'BackgroundColor',colorfondo,...
    'FontSize',14,...
    'ForegroundColor',colortxt,...
    'Horizontalalignment','right',...
    'Position',[0.2 0.3 0.6 0.2],...
    'String',integrantes,...
    'Style','text');

% Texto "Director" ==============
uicontrol(...
    'Units',uni,...
    'BackgroundColor',colorfondo,...
    'FontSize',14,...
    'ForegroundColor',colortxt,...
    'Horizontalalignment','right',...
    'Position',[0.2 0.15 0.6 0.1],...
    'String','Director:  Ing. Hinojosa Miguel',...
    'Style','text');
set(figini,'Visible','on');                    % Pantalla Visible
pause(5)
close;
GrafCONF;      % Llama a la pantalla configurar Tarjetas
```

GRAFCONF

```
% ========== Grafico CONFIGURACION ========== %
function GrafCONF
global datos
datos=[];
colorfondo=[0.9255 0.9137 0.8471];
colortxt=[0.3 0.3 1];
uni='Normalized';
datos.pos.CONF=[0.3 0.3 0.4 0.4];
if exist('configuracion.mat')==0
   [tarjetas1]=daqgate('privateGetAdaptor','analoginput');
   [tarjetas2]=daqgate('privateGetAdaptor','analogoutput');
   [tarjetas3]=daqgate('privateGetAdaptor','digitalio');
   save('configuracion.mat','tarjetas1','tarjetas2','tarjetas3','-mat');
else
   carga=load('configuracion.mat','-mat');
   [tarjetas1]=carga.tarjetas1;
   [tarjetas2]=carga.tarjetas2;
   [tarjetas3]=carga.tarjetas3;
end
datos.conf=[];
datos.dac=[];
datos.dio=[];
datos.ref=0;

% Figura y Elementos ==============
figconf=figure(...
   'Units',uni,...
   'Color',colorfondo,...
   'Menubar','none',...
   'Name','Configurar',...
   'NumberTitle','off',...
```

```
    'Position',datos.pos.CONF,...
    'Resize','off',...
    'ToolBar','none',...
    'Visible','off',...
    'WindowStyle','Modal');

% Texto "Configuracion de las tarjetas a utilizar" ==============
uicontrol(...
    'BackgroundColor',colorfondo,...
    'FontAngle','normal',...
    'FontName','Arial',...
    'FontSize',14,...
    'ForegroundColor',colortxt,...
    'Horizontalalignment','center',...
    'Units',uni,...
    'Position',[0.05 0.75 0.9 0.2],...
    'String','Configuracion de las tarjetas',...
    'Style','text');

% Texto "Configuracion de las entradas analogicas" ==============
uicontrol(...
    'BackgroundColor',colorfondo,...
    'FontAngle','normal',...
    'FontName','Arial',...
    'FontSize',9,...
    'ForegroundColor',colortxt,...
    'Horizontalalignment','left',...
    'Units',uni,...
    'Position',[0.25 0.6 0.2 0.12],...
    'String','Entradas Analogicas',...
    'Style','text');

% ========= ComboBox "Entradas Analogicas" =========
```

```
datos.conf.conf1=uicontrol(...
   'Units',uni,...
   'BackgroundColor',colorfondo,...
   'HorizontalAlignment','left',...
      'Position',[0.45 0.65 0.25 0.05],...
      'String',tarjetas1,...
   'Style','popup');

% Texto "Configuracion de las salidas analogicas" ==============
uicontrol(...
   'BackgroundColor',colorfondo,...
   'FontAngle','normal',...
   'FontName','Arial',...
   'FontSize',9,...
   'ForegroundColor',colortxt,...
   'Horizontalalignment','left',...
   'Units',uni,...
   'Position',[0.25 0.43 0.2 0.12],...
   'String','Salidas Analogicas',...
   'Style','text');

% ========= ComboBox "Salidas Analogicas" =========
datos.conf.conf2=uicontrol(...
   'Units',uni,...
   'BackgroundColor',colorfondo,...
   'HorizontalAlignment','left',...
      'Position',[0.45 0.48 0.25 0.05],...
      'String',tarjetas2,...
   'Style','popup');

% Texto "Configuracion de las entradas / salidas digitales" ==============
uicontrol(...
   'BackgroundColor',colorfondo,...
```

```
    'FontAngle','normal',...
    'FontName','Arial',...
    'FontSize',9,...
    'ForegroundColor',colortxt,...
    'Horizontalalignment','left',...
    'Units',uni,...
    'Position',[0.25 0.23 0.2 0.12],...
    'String','Entra/Sali Digitales',...
    'Style','text');

% ========= ComboBox "Entradas / Salidas Digitales" =========
datos.conf.conf3=uicontrol(...
    'Units',uni,...
    'BackgroundColor',colorfondo,...
    'HorizontalAlignment','left',...
        'Position',[0.45 0.28 0.25 0.05],...
        'String',tarjetas3,...
    'Style','popup');

% Botón "Continuar" ===============
uicontrol(...
    'Style','push',...
    'Units',uni,...
    'BackgroundColor','cyan',...
    'Fontsize',8,...
    'ForegroundColor',colortxt,...
    'Position',[0.41 0.05 0.18 0.08],...
    'Callback','CerrarCONF',...
    'String','CONTINUAR');
set(figconf,'Visible','on');                    % Pantalla Visible
```

CERRARCONF

```
% ========== CERRAR CONFIGURACION TARJETAS ========== %
function CerrarCONF
global datos
val1=get(datos.conf.conf1,'Value');
str1=get(datos.conf.conf1,'String');
val2=get(datos.conf.conf2,'Value');
str2=get(datos.conf.conf2,'String');
val3=get(datos.conf.conf3,'Value');
str3=get(datos.conf.conf3,'String');
if iscell(str1)
   daqStr1=str1{val1};
end
if iscell(str2)
   daqStr2=str2{val2};
end
if iscell(str3)
   daqStr3=str3{val3};
end
espacio1=find(daqStr1==' ');
espacio2=find(daqStr2==' ');
espacio3=find(daqStr3==' ');
if isempty(espacio1)
   msgbox('No tiene tarjeta de Entradas analogicas. Se cerrara el
programa','Error','error')
   close('gcf');
elseif isempty(espacio2)
   msgbox('No tiene tarjeta de Salidas analogicas. Se cerrara el
programa','Error','error')
   close('gcf');
elseif isempty(espacio3)
```

```
   msgbox('No tiene tarjeta de Entradas/Salidas digitales. Se cerrara el
programa','Error','error')
   close('gcf');
else
   datos.dac.daq1=daqStr1(1:espacio1-1);
   datos.dac.id1=daqStr1(espacio1+1:end);
   can1=daqhwinfo(analoginput(datos.dac.daq1,datos.dac.id1));
   datos.dac.daq2=daqStr2(1:espacio2-1);
   datos.dac.id2=daqStr2(espacio2+1:end);
   can2=daqhwinfo(analogoutput(datos.dac.daq2,datos.dac.id2));
   datos.dac.daq3=daqStr3(1:espacio3-1);
   datos.dac.id3=daqStr3(espacio3+1:end);
   can3=daqhwinfo(digitalio(datos.dac.daq3,datos.dac.id3));
   if can1.TotalChannels<8
      msgbox('Tarjeta no tiene el numero suficiente de Entradas analogicas para
ejecutar programa','Error','error')
   elseif can2.TotalChannels<2
      msgbox('Tarjeta no tiene el numero suficiente de Salidas analogicas para
ejecutar programa','Error','error')
   elseif can3.TotalLines<24
      msgbox('Tarjeta no tiene el numero suficiente de Entradas/Salidas
analogicas para ejecutar programa','Error','error')
   else
      close;
      controlador;
   end
end
```

CONTROLADOR

```
% ========== Programa Principal ========== %
function Controlador
coeficientes;      %Coeficientes para los filtros.
```

```
digital;        %Creacion de las entradas-salidas digitales.
analogico;      %Creacion de las entradas analogicas.
Grafico;        %Creacion del interfaz de usuario.
```

COEFICIENTES

```
function coeficientes
global datos
datos.Num1=[0.099564306302893  -0.398093004959703  0.597057413580654  -
0.398093004959703 0.099564306302893];
datos.Den1=[1.000000000000000  -3.989827426102192  5.970386749947586  -
3.971287148084355 0.990727986894077];
datos.Num2=[0.234367946945844  -0.571235248455397  0.499826684482913  -
0.606921088200406  0.887923547132343  -0.606921088200407
0.499826684482914 -0.571235248455398 0.234367946945845];
datos.Den2=[1.000000000000000  -4.188996762017442  8.004492217889990  -
10.301174613315027  10.317178707320686  -7.645814936021341
3.951492399075484 -1.432303695611604 0.295128049761257];
datos.Num3=[0.106469385462474  0.336402454184126  0.563146783887494
0.636513293464167  0.454940820173147  0.000000000000004  -
0.454940820173139  -0.636513293464160  -0.563146783887489  -
0.336402454184124 -0.106469385462474];
datos.Den3=[1.000000000000000  4.107541668351901  10.044701658271228
17.168594861959061  22.731186954276094  23.514473717350178
19.456595376341092  12.568942460825333  6.236759866487948
2.117564796395137 0.431709115812180];
datos.barr1=1;
datos.barr2=1;
datos.barr3=1;
```

DIGITAL

```
% ========== Entradas/Salidas Digitales ========== %
function Digital
global datos
datos.dio=digitalio(datos.dac.daq3,datos.dac.id3);
addline(datos.dio,0:23,'Out');
putvalue(datos.dio.line(5),0);
putvalue(datos.dio.line(8),0);
putvalue(datos.dio.line(13),0);
putvalue(datos.dio.line(16),0);
putvalue(datos.dio.line(22),1);  %Señal a adquirir de 5.12 V
```

ANALOGICO

```
% ========== Entradas Analogicas ========== %
function analogico
global datos
datos.ai=analoginput(datos.dac.daq1,datos.dac.id1);
addchannel(datos.ai,0:7);
datos.ao=analogoutput(datos.dac.daq2,datos.dac.id2);
addchannel(datos.ao,1:2);
set(datos.ai,'InputType','Single');
datos.amplitud=[-1.25 1.25];
for i=1:8
   set(datos.ai.Channel(i),'InputRange',datos.amplitud);
   set(datos.ai.Channel(i),'SensorRange',datos.amplitud);
   set(datos.ai.Channel(i),'UnitsRange',datos.amplitud);
end
set(datos.ao,'StandardSampleRates','Off');
set([datos.ai datos.ao],'SampleRate',31250);
set([datos.ai datos.ao],'TriggerType','Manual');
```

```
set(datos.ai,'ManualTriggerHwOn','Trigger');
set(datos.ai,'SamplesPerTrigger',inf);
datos.muestras=3905;
%datos.muestras=2500;
valores=zeros(4*datos.muestras,2);
putdata(datos.ao,valores);
pru=datos.ai;
set(datos.ao,'SamplesOutputAction',{'actualizar',pru});
set(datos.ao,'SamplesOutputActionCount',datos.muestras);
set([datos.ai datos.ao],'StopAction','daqaction');
start([datos.ai datos.ao]);
trigger([datos.ai datos.ao]);
datos.tiempo=zeros(datos.muestras,8);
datos.bar=[];
datos.barra=[];
datos.cheq=0;
datos.det=0;
datos.pantalla=[];
datos.pun=[];
datos.texto=[];
datos.val=[];
```

ACTUALIZAR

```
% ========== Llamado a funcion ACTUALIZAR ========== %
function actualizar(obj,event,pru)
global datos
eje=datos.pantalla.ejes;
linea=datos.pantalla.lineas;
datos.vec=[get(datos.pun.pun1,'Value')                    get(datos.pun.pun2,'Value')
get(datos.pun.pun3,'Value') get(datos.pun.pun4,'Value')];
datos.tiempo=getdata(pru,datos.muestras);
```

```
datos.canal=[datos.tiempo(:,3)*get(datos.bar.bar1,'Value')
datos.tiempo(:,1)*get(datos.bar.bar2,'Value')
datos.tiempo(:,7)*get(datos.bar.bar3,'Value')
datos.tiempo(:,5)*get(datos.bar.bar4,'Value')];
datos.mezcla=[datos.tiempo(:,1)*get(datos.bar.bar2,'Value')+datos.tiempo(:,3)*get(
datos.bar.bar1,'Value')+datos.tiempo(:,5)*get(datos.bar.bar4,'Value')+datos.tiempo
(:,7)*get(datos.bar.bar3,'Value')...

datos.tiempo(:,1)*get(datos.bar.bar2,'Value')+datos.tiempo(:,2)*get(datos.bar.bar1,
'Value')+datos.tiempo(:,4)*get(datos.bar.bar4,'Value')+datos.tiempo(:,6)*get(datos.
bar.bar3,'Value')];
datos.valor=[datos.mezcla(:,1)*get(datos.bar.bar5,'Value')
datos.mezcla(:,2)*get(datos.bar.bar6,'Value')];
datos.filtrar(:,1)=datos.barr1*filter(datos.Num1,datos.Den1,datos.valor(:,1))+datos.
barr2*filter(datos.Num2,datos.Den2,datos.valor(:,1))+datos.barr3*filter(datos.Num
3,datos.Den3,datos.valor(:,1));
datos.filtrar(:,2)=datos.barr1*filter(datos.Num1,datos.Den1,datos.valor(:,2))+datos.
barr2*filter(datos.Num2,datos.Den2,datos.valor(:,2))+datos.barr3*filter(datos.Num
3,datos.Den3,datos.valor(:,2));
if datos.ref==0
   impedancia;
   datos.ref=1;
end
datos.Vx1=max(datos.filtrar(:,1));
datos.Vx2=max(datos.filtrar(:,2));
switch round(datos.imped)
case 2
   if datos.Vx1>0.186
      datos.filtrar(:,1)=(0.186/datos.Vx1)*datos.filtrar(:,1);
   end
   if datos.Vx2>0.186
      datos.filtrar(:,2)=(0.186/datos.Vx2)*datos.filtrar(:,2);
   end
```

```
case 4
   if datos.Vx1>0.263
      datos.filtrar(:,1)=(0.263/datos.Vx1)*datos.filtrar(:,1);
   end
   if datos.Vx2>0.263
      datos.filtrar(:,2)=(0.263/datos.Vx2)*datos.filtrar(:,2);
   end
case 6
   if datos.Vx1>0.322
      datos.filtrar(:,1)=(0.322/datos.Vx1)*datos.filtrar(:,1);
   end
   if datos.Vx2>0.322
      datos.filtrar(:,2)=(0.322/datos.Vx2)*datos.filtrar(:,2);
   end

case 8
   if datos.Vx1>0.372
      datos.filtrar(:,1)=(0.372/datos.Vx1)*datos.filtrar(:,1);
   end
   if datos.Vx2>0.372
      datos.filtrar(:,2)=(0.372/datos.Vx2)*datos.filtrar(:,2);
   end
case 16
   if datos.Vx1>0.526
      datos.filtrar(:,1)=(0.526/datos.Vx1)*datos.filtrar(:,1);
   end
   if datos.Vx2>0.526
      datos.filtrar(:,2)=(0.526/datos.Vx2)*datos.filtrar(:,2);
   end
end
putdata(obj,datos.filtrar);
potencia;
```

```
%EJE 1 ++++++++++++++++++++++
if datos.vec(3)==1
   set(eje(1),'XLim',[1 datos.muestras]);
   if datos.vec(1)<=2
      set(linea(1),'YData',datos.filtrar(:,datos.vec(1)));
   else
      set(linea(1),'YData',datos.canal(:,datos.vec(1)-2));
   end
else
   [f,mag]=Frecuencia(1);
   maxF=max(f);
   minF=min(f);
   xax=get(eje(1),'XLim');
   if minF<xax(1)
      xax(1)=minF;
   end
   if maxF>xax(2)
      xax(2)=maxF;
   end
   set(eje(1),'XLim',xax)
   maxM=max(mag);
   minM=min(mag);
   yax1=get(eje(1),'YLim');
   if minM<yax1(1)
      yax1(1)=minM;
   end
   if maxM>yax1(2)
      yax1(2)=maxM;
   end
   set(eje(1),'YLim',yax1)
   set(linea(1),'XData',f,'YData',mag);
end
```

```
%EJE 2 ++++++++++++++++++++++
if datos.vec(4)==1
   set(eje(2),'XLim',[1 datos.muestras]);
   if datos.vec(2)<=2
      set(linea(2),'YData',datos.filtrar(:,datos.vec(2)));
   else
      set(linea(2),'YData',datos.canal(:,datos.vec(2)-2));
   end
else
   [f,mag]=Frecuencia(2);
   maxF=max(f);
   minF=min(f);
   xax=get(eje(2),'XLim');
   if minF<xax(1)
      xax(1)=minF;
   end
   if maxF>xax(2)
      xax(2)=maxF;
   end
   set(eje(2),'XLim',xax);
   maxM=max(mag);
   minM=min(mag);
   yax1=get(eje(2),'YLim');
   if minM<yax1(1)
      yax1(1)=minM;
   end
   if maxM>yax1(2)
      yax1(2)=maxM;
   end
   set(eje(2),'YLim',yax1);
   set(linea(2),'XData',f,'YData',mag);
end
drawnow;
```

GRAFICO

```
% ========== Figura y Elementos ========== %
function Grafico
global datos
tam=get(0,'ScreenSize');
posicionfig=[tam(1) tam(2) tam(3) tam(4)-50];
uni='normalized';
colorfig=[0.9255 0.9137 0.8471];
colortxt=[0.3 0.3 1];
% Pantalla principal =========
datos.pantalla.figura=figure(...
   'Color',colorfig,...
   'Menubar','none',...
   'Name','Pantalla Principal',...
   'NumberTitle','off',...
   'Position',posicionfig,...
   'Resize','off',...
      'ToolBar','none',...
   'UserData',[],...
   'Visible','off');
ab=0.8;                                    %posición inferior (Abajo)
espab=0.04;                              %espacio entre textos (hacia Abajo)
texto(uni,colorfig,colortxt,ab,espab);          %Llamado a la funcción TEXTO
minimo=0;
maximo=10;
pasos=[0.1/(maximo-minimo) 0.01/(maximo-minimo)];
valor=0;

% Barra desplazadora "MICRÓFONO 1" =========
datos.bar.bar1=uicontrol(...
   'Units',uni,...
```

```
        'BackgroundColor',colorfig,...
    'Callback','Potenciometro("pot1")',...
    'Position',[(0.03+2.25*espab) ab 0.15 0.03],...
    'Min',minimo,...
    'Max',maximo,...
    'SliderStep',pasos,...
    'Value',valor,...
    'Style','slider');

% ========= Barra desplazadora "MICRÓFONO 2" =========
datos.bar.bar2=uicontrol(...
    'Units',uni,...
    'BackgroundColor',colorfig,...
    'Callback','Potenciometro("pot2")',...
        'Position',[(0.03+2.25*espab) (ab-espab) 0.15 0.03],...
    'Min',minimo,...
    'Max',maximo,...
    'SliderStep',pasos,...
    'Value',valor,...
    'Style','slider');

% ========= Barra desplazadora "AUXILIAR 1" =========
datos.bar.bar3=uicontrol(...
    'Units',uni,...
    'BackgroundColor',colorfig,...
    'Callback','Potenciometro("pot3")',...
        'Position',[(0.03+2.25*espab) (ab-2*espab) 0.15 0.03],...
    'Min',minimo,...
    'Max',maximo,...
    'SliderStep',pasos,...
    'Value',valor,...
        'Style','slider');
```

```
% ========= Barra desplazadora "AUXILIAR 2" =========
datos.bar.bar4=uicontrol(...
        'Units',uni,...
        'BackgroundColor',colorfig,...
    'Callback','Potenciometro("pot4")',...
        'Position',[(0.03+2.25*espab) (ab-3*espab) 0.15 0.03],...
    'Min',minimo,...
    'Max',maximo,...
    'SliderStep',pasos,...
    'Value',valor,...
        'Style','slider');

% ========= Barra desplazadora "MASTER A" =========
datos.bar.bar5=uicontrol(...
        'Units',uni,...
        'BackgroundColor',colorfig,...
    'Callback','Potenciometro("pot5")',...
        'Position',[(0.03+2.25*espab) (ab-4*espab) 0.15 0.03],...
    'Min',minimo,...
    'Max',maximo,...
    'SliderStep',pasos,...
    'Value',valor,...
        'Style','slider');

% ========= Barra desplazadora "MASTER B" =========
datos.bar.bar6=uicontrol(...
        'Units',uni,...
        'BackgroundColor',colorfig,...
    'Callback','Potenciometro("pot6")',...
        'Position',[(0.03+2.25*espab) (ab-5*espab) 0.15 0.03],...
    'Min',minimo,...
    'Max',maximo,...
    'SliderStep',pasos,...
```

```
    'Value',valor,...
        'Style','slider');

ancho=0.05;          %ancho de los VAL
datos.valores=[get(datos.bar.bar1,'Value')                    get(datos.bar.bar2,'Value')
get(datos.bar.bar3,'Value') get(datos.bar.bar4,'Value')]
% ========= Valor "MICRÓFONO 1" =========
datos.val.val1=uicontrol(...
        'Units',uni,...
        'BackgroundColor',colorfig,...
        'ForegroundColor',colortxt,...
    'HorizontalAlignment','left',...
        'Position',[(0.03+6.2*espab) ab ancho 0.03],...
        'String',datos.valores(1),...
        'Style','text');

% ========= Valor "MICRÓFONO 2" =========
datos.val.val2=uicontrol(...
        'Units',uni,...
    'BackgroundColor',colorfig,...
        'ForegroundColor',colortxt,...
        'HorizontalAlignment','left',...
        'Position',[(0.03+6.2*espab) (ab-espab) ancho 0.03],...
        'String',datos.valores(2),...
        'Style','text');

% ========= Valor "AUXILIAR 1" =========
datos.val.val3=uicontrol(...
        'Units',uni,...
        'BackgroundColor',colorfig,...
        'ForegroundColor',colortxt,...
    'HorizontalAlignment','left',...
        'Position',[(0.03+6.2*espab) (ab-2*espab) ancho 0.03],...
```

```
        'String',datos.valores(3),...
        'Style','text');

% ========= Valor "AUXILIAR 2" =========
datos.val.val4=uicontrol(...
        'Units',uni,...
        'BackgroundColor',colorfig,...
        'ForegroundColor',colortxt,...
    'HorizontalAlignment','left',...
        'Position',[(0.03+6.2*espab) (ab-3*espab) ancho 0.03],...
        'String',datos.valores(4),...
        'Style','text');

% ========= Valor "MASTER A" =========
datos.val.val5=uicontrol(...
        'Units',uni,...
        'BackgroundColor',colorfig,...
        'ForegroundColor',colortxt,...
    'HorizontalAlignment','left',...
        'Position',[(0.03+6.2*espab) (ab-4*espab) ancho 0.03],...
        'String',get(datos.bar.bar5,'Value'),...
        'Style','text');

% ========= Valor "MASTER B" =========
datos.val.val6=uicontrol(...
        'Units',uni,...
        'BackgroundColor',colorfig,...
        'ForegroundColor',colortxt,...
    'HorizontalAlignment','left',...
        'Position',[(0.03+6.2*espab) (ab-5*espab) ancho 0.03],...
        'String',get(datos.bar.bar6,'Value'),...
        'Style','text');
```

```
% ========= ComboBox "Punto 1" =========
combo1=str2mat('Salida A','Salida B','Micrófono 1','Micrófono 2','Auxiliar 1','Auxiliar
2');
datos.pun.pun1=uicontrol(...
   'Units',uni, ...
   'BackgroundColor',colorfig,...
   'Callback','Salida("salida1")',...
   'HorizontalAlignment','left',...
      'Position',[0.095 (ab-12*espab) 0.11 0.04],...
      'String',combo1,...
   'Style','popup');

% ========= ComboBox "Punto 2" =========
datos.pun.pun2=uicontrol(...
      'Units',uni,...
      'BackgroundColor',colorfig,...
      'Callback','Salida("salida2")',...
   'HorizontalAlignment','left',...
      'Position',[0.095 (ab-13*espab) 0.11 0.04],...
      'String',combo1,...
   'Style','popup',...
   'Value',2);

% ========= ComboBox "Dominio 1" =========
combo2=str2mat('Tiempo','Frecuencia');
datos.pun.pun3=uicontrol(...
   'Units',uni,...
      'BackgroundColor',colorfig,...
   'Callback','Dominio("dominio1")',...
   'HorizontalAlignment','left',...
      'Position',[0.095 (ab-14*espab) 0.11 0.04],...
      'String',combo2,...
   'Style','popup');
```

```
% ========= ComboBox "Dominio 2" =========
datos.pun.pun4=uicontrol(...
   'Units',uni,...
   'BackgroundColor',colorfig,...
   'Callback','Dominio("dominio2")',...
   'HorizontalAlignment','left',...
      'Position',[0.095 (ab-15*espab) 0.11 0.04],...
      'String',combo2,...
   'Style','popup');

% ========= Botón "Detener" =========
datos.pantalla.detener=uicontrol(...
      'Units',uni,...
      'BackgroundColor',[0.91 0.91 0.91],...
      'Callback','Detener',...
   'Fontsize',8,...
      'ForegroundColor',colortxt,...
   'Parent',datos.pantalla.figura,...
      'Position',[0.08 (ab-16.5*espab) 0.14 0.04],...
      'String','Detener Adquisicion',...
      'Style','push');

% ========= Botón "Salir" =========
uicontrol(...
      'Units',uni,...
      'BackgroundColor',[0.91 0.91 0.91],...
      'Callback','Cerrar',...
   'Fontsize',12,...
      'ForegroundColor',colortxt,...
   'Parent',datos.pantalla.figura,...
      'Position',[0.095 (ab-18*espab) 0.11 0.04],...
      'String','SALIR',...
```

```
        'Style','push');

% ========= Eje 1 =========
izq=0.38;               %posición izquierda de los ejes
eje(1)=axes(...
    'Position',[izq 0.55 0.60 0.35],...
    'Parent',datos.pantalla.figura,...
    'XGrid','On',...
    'XLim',[1 datos.muestras],...
    'XLimMode','Manual',...
    'YGrid','On',...
    'Ylim',datos.amplitud,...
    'YLimMode','manual');
linea(1)=line(...
    'Color',colortxt,...
    'Parent',eje(1),...
    'XData',1:datos.muestras,...
    'YData',datos.tiempo(:,1));

% ========= Eje 2 =========
eje(2)=axes(...
    'Position',[izq 0.10 0.60 0.35],...
    'Parent',datos.pantalla.figura,...
    'XGrid','On',...
    'XLim',[1 datos.muestras],...
    'XLimMode','Manual',...
    'YGrid','On',...
    'Ylim',datos.amplitud,...
    'YLimMode','manual');
linea(2)=line(...
    'Color',colortxt,...
    'Parent',eje(2),...
    'XData',1:datos.muestras,...
```

```
    'YData',datos.tiempo(:,1));

%          ========================          Creación          de          Menú
========================
Archivo=uimenu('Parent',datos.pantalla.figura,...
   'Label','Archivo');
Cerra=uimenu(Archivo,...
   'Label','Salir',...
   'Callback','Cerrar',...
   'Separator','On',...
   'Accelerator','S');
Frase=uimenu('Parent',datos.pantalla.figura,...
   'Label','Funciones');
datos.ecualiza=uimenu(Frase,...
   'Label','Ecualizador',...
   'Callback','GrafEQ',...
   'Accelerator','E');
datos.detener=uimenu(Frase,...
   'Label','Iniciar',...
   'Callback','Detener',...
   'Separator','On',...
   'Accelerator','D');
datos.impedancia=uimenu(Frase,...
   'Label','Impedancia',...
   'Callback','Impedancia',...
   'Accelerator','P');
datos.inhabilita=uimenu(Frase,...
   'Label','Inhabilitar Controles',...
   'Callback','Inhabilitar',...
   'Accelerator','I');
Ayu=uimenu('Parent',datos.pantalla.figura,...
   'Label','Ayuda');
Ayu0=uimenu(Ayu,...
```

```
  'Label','Ayuda',...
  'Callback','Ayuda("AyudaA")',...
  'Accelerator','H');
Ayu01=uimenu(Ayu,...
  'Label','Ayuda General',...
  'Separator','On');
Ayud3=uimenu(Ayu01,...
  'Label','Codigo',...
  'Callback','Ayuda("AyudaB")');
Ayud1=uimenu(Ayu01,...
  'Label','Conceptos',...
  'Callback','Ayuda("AyudaC")');
Ayud2=uimenu(Ayu01,...
  'Label','Funcionamiento',...
  'Callback','Ayuda("AyudaF")');
Ayud4 = uimenu(Ayu01,...
  'Label','Informacion',...
  'Callback','Ayuda("Ayudal")');
set(datos.pantalla.figura,'Visible','on');
datos.pantalla.ejes=eje;
datos.pantalla.lineas=linea;
```

TEXTO

```
% ========== Funcion texto de la pantalla ========== %
function texto(uni,colorfig,colortxt,ab,espab)
global datos
% ========= Texto "CONTROLADOR" =========
uicontrol('Units',uni,...
   'BackgroundColor',colorfig,...
   'FontName','Arial',...
   'FontSize',22,...
```

```
    'FontWeight','demi',...
    'ForegroundColor',colortxt,...
    'Position',[0.35 0.94 0.3 0.06],...
    'String','CONTROLADOR',...
    'Value',1,...
    'Enable','Inactive',...
    'Style','togglebutton');

% ====== Texto 'Micrófono 1','Micrófono 2','Auxiliar 1','Auxiliar 2','Master
A','Master B'======

postxt={ab,ab-espab,ab-2*espab,ab-3*espab,ab-4*espab,ab-5*espab};
strtxt={'Micrófono 1','Micrófono 2','Auxiliar 1','Auxiliar 2','Master A','Master B'};
for i=1:6
    uicontrol('Units',uni,...
        'HorizontalAlignment','left',...
        'BackgroundColor',colorfig,...
        'ForegroundColor',colortxt,...
        'Position',[0.03 postxt{i} 0.11 0.03],...
        'String',strtxt{i},...
        'Style','text');
end

% ========= Texto 'Potencia de Salida','Resistencia de Carga' =========
postxt={ab-8*espab,ab-9*espab};
strtxt={'Potencia de Salida =','Resistencia de Carga = '};
for i=1:2
    uicontrol('Units',uni,...
        'HorizontalAlignment','left',...
        'BackgroundColor',colorfig,...
        'ForegroundColor',colortxt,...
        'Position',[0.03 postxt{i} 0.14 0.03],...
        'String',strtxt{i},...
```

```
        'Style','text');
end

% ========= Texto "Valor Potencia de Salida" =========
datos.potencia.pot=uicontrol('Units',uni,...
    'HorizontalAlignment','left',...
    'BackgroundColor',colorfig,...
    'ForegroundColor',colortxt,...
    'Position',[0.17 (ab-8*espab) 0.105 0.03],...
    'String','_____(vatios)',...
    'Style','text');

% ========= Texto "Valor Resistencia de Carga" =========
datos.potencia.res=uicontrol('Units',uni,...
    'HorizontalAlignment','left',...
    'BackgroundColor',colorfig,...
    'ForegroundColor',colortxt,...
    'Position',[0.17 (ab-9*espab) 0.105 0.03],...
    'String','_____(ohm)',...
    'Style','text');

% ========= Texto "Puntos y Dominios a Graficar" =========
uicontrol('Units',uni,...
    'BackgroundColor',colorfig,...
    'ForegroundColor',colortxt,...
    'Position',[0.035 (ab-11*espab+0.01) 0.17 0.03],...
    'String','Puntos y Dominio a Graficar',...
    'Style','text');

% ========= Texto "Punto 1" =========
postxt={ab-12*espab,ab-13*espab,ab-14*espab,ab-15*espab};
strtxt={'Punto 1:','Punto 2:','Dominio 1:','Dominio 2:'};
for i=1:4
```

```
   uicontrol('Units',uni,...
      'HorizontalAlignment','left',...
      'BackgroundColor',colorfig,...
      'ForegroundColor',colortxt,...
      'Position',[0.03 postxt{i} 0.14 0.03],...
      'String',strtxt{i},...
      'Style','text');
end

% ========= Línea separadora "LINE" =========
uicontrol('Units',uni,...
   'BackgroundColor',colortxt,...
   'Position',[0.03 (ab-5.6*espab) 0.25 0.005],...
   'Style','frame');
```

CERRAR

```
% ========== Llamado a funcion Cerrar ========== %
function Cerrar
global datos
putvalue(datos.dio.line(7),0);
putvalue(datos.dio.line(10),0);
putvalue(datos.dio.line(11),0);
putvalue(datos.dio.line(14),0);
for i=0.1:0.1:2*datos.valores(1)
   k=1;
   if getvalue(datos.dio.line(k))==0          % Contador (INC)
      putvalue(datos.dio.line(k),1);
      putvalue(datos.dio.line(k+2),1);
   else
      putvalue(datos.dio.line(k),0);
```

```
      putvalue(datos.dio.line(k+2),0);
   end
end
for i=0.1:0.1:2*datos.valores(2)
   k=4;
   if getvalue(datos.dio.line(k))==0          % Contador (INC)
      putvalue(datos.dio.line(k),1);
      putvalue(datos.dio.line(k+2),1);
   else
      putvalue(datos.dio.line(k),0);
      putvalue(datos.dio.line(k+2),0);
   end
end
for i=0.1:0.1:2*datos.valores(3)
   k=15;
   if getvalue(datos.dio.line(k))==0          % Contador (INC)
      putvalue(datos.dio.line(k),1);
      putvalue(datos.dio.line(k+2),1);
   else
      putvalue(datos.dio.line(k),0);
      putvalue(datos.dio.line(k+2),0);
   end
end
for i=0.1:0.1:2*datos.valores(4)
   k=18;
   if getvalue(datos.dio.line(k))==0          % Contador (INC)
      putvalue(datos.dio.line(k),1);
      putvalue(datos.dio.line(k+2),1);
   else
      putvalue(datos.dio.line(k),0);
      putvalue(datos.dio.line(k+2),0);
   end
end
```

```
if isvalid(datos.ai)
   if strcmp(get(datos.ai,'Running'),'On')      % Comparacion de String´s
      stop([datos.ai datos.ao]);
   end
   delete([datos.ai datos.ao datos.dio]);
end
close all
clc
```

GRAFEQ

```
% ========== Grafico ECUALIZADOR ========== %
function GrafEQ
global datos
colorfondo=[0.9255 0.9137 0.8471];
colortxt=[0.3 0.3 1];
uni='Normalized';
posicionfig=[0.8 0.1 0.15 0.25];
alto=0.28;
ancho=0.1;
espancho=0.175;
espalto=0.8;
espabajo=0.18;
espa=0.2;

% Figura y Elementos ==============
figeq=figure(...
   'Color',colorfondo,...
   'Units',uni,...
   'Menubar','none',...
   'Name','Ecualizador',...
   'NumberTitle','off',...
```

```
    'Position',posicionfig,...
    'ToolBar','none',...
    'Resize','off',...
    'Visible','off');

% Texto "FRECUENCIA 1" ===============
uicontrol(...
    'BackgroundColor',colorfondo,...
    'FontAngle','normal',...
    'FontName','Arial',...
    'FontSize',8,...
    'ForegroundColor',colortxt,...
    'Horizontalalignment','center',...
    'Units',uni,...
    'Position',[(espancho-0.02) espalto espa 0.08],...
    'String','100',...
    'Style','text');

% Texto "FRECUENCIA 2" ===============
uicontrol(...
    'BackgroundColor',colorfondo,...
    'FontAngle','normal',...
    'FontName','Arial',...
    'FontSize',8,...
    'ForegroundColor',colortxt,...
    'Horizontalalignment','center',...
    'Units',uni,...
    'Position',[(2*espancho+ancho-0.02) espalto espa 0.08],...
    'String','1K',...
    'Style','text');

% Texto "FRECUENCIA 3" ===============
uicontrol(...
```

```
    'BackgroundColor',colorfondo,...
    'FontAngle','normal',...
    'FontName','Arial',...
    'FontSize',8,...
    'ForegroundColor',colortxt,...
    'Horizontalalignment','center',...
    'Units',uni,...
    'Position',[(3*espancho+2*ancho-0.02) espalto espa 0.08],...
    'String','10K',...
    'Style','text');

minimo=-3;
maximo=3;
pasos=[1/(maximo-minimo) 0.18/(maximo-minimo)];
valor=0;

% Barra desplazadora "FRECUENCIA 1" =========
datos.barra.barra1=uicontrol(...
        'BackgroundColor',colorfondo,...
    'Callback','Ecualizador("eq1")',...
    'Min',minimo,...
    'Max',maximo,...
    'SliderStep',pasos,...
    'Value',valor,...
    'Units',uni,...
    'Position',[espancho alto ancho 0.5],...
    'Style','slider');

% Barra desplazadora "FRECUENCIA 2" =========
datos.barra.barra2=uicontrol(...
        'BackgroundColor',colorfondo,...
    'Callback','Ecualizador("eq2")',...
    'Min',minimo,...
```

```
    'Max',maximo,...
    'SliderStep',pasos,...
    'Value',valor,...
    'Units',uni,...
    'Position',[(2*espancho+ancho) alto ancho 0.5],...
    'Style','slider');

% Barra desplazadora "FRECUENCIA 3" =========
datos.barra.barra3=uicontrol(...
        'BackgroundColor',colorfondo,...
    'Callback','Ecualizador("eq3")',...
    'Min',minimo,...
    'Max',maximo,...
    'SliderStep',pasos,...
    'Value',valor,...
    'Units',uni,...
    'Position',[(3*espancho+2*ancho) alto ancho 0.5],...
    'Style','slider');

% Texto "VALOR 1" ===============
datos.texto.texto1=uicontrol(...
    'BackgroundColor',colorfondo,...
    'FontAngle','normal',...
    'FontName','Arial',...
    'FontSize',8,...
    'ForegroundColor',colortxt,...
    'Horizontalalignment','center',...
    'Units',uni,...
    'Position',[(espancho-0.02) espabajo espa 0.08],...
    'String',get(datos.barra.barra1,'Value'),...
    'Style','text');

 % Texto "VALOR 2" ===============
```

```
datos.texto.texto2=uicontrol(...
    'BackgroundColor',colorfondo,...
    'FontAngle','normal',...
    'FontName','Arial',...
    'FontSize',8,...
    'ForegroundColor',colortxt,...
    'Horizontalalignment','center',...
    'Units',uni,...
    'Position',[(2*espancho+ancho-0.02) espabajo espa 0.08],...
    'String',get(datos.barra.barra2,'Value'),...
    'Style','text');

% Texto "VALOR 3" ===============
datos.texto.texto3=uicontrol(...
    'BackgroundColor',colorfondo,...
    'FontAngle','normal',...
    'FontName','Arial',...
    'FontSize',8,...
    'ForegroundColor',colortxt,...
    'Horizontalalignment','center',...
    'Units',uni,...
    'Position',[(3*espancho+2*ancho-0.02) espabajo espa 0.08],...
    'String',get(datos.barra.barra3,'Value'),...
    'Style','text');

% Botón "Cerrar" ===============
uicontrol(...
    'Style','push',...
    'Units',uni,...
    'BackgroundColor','white',...
    'Fontsize',8,...
    'ForegroundColor',colortxt,...
    'Position',[0.33 0.02 0.34 0.1],...
```

```
   'Callback','close(gcf)',...
   'String','CERRAR');
set(figeq,'Visible','on');                    % Pantalla Visible
```

POTENCIOMETRO

```
% ========== Llamado a funcion POTENCIOMETRO ========== %
function Potenciometro(evento)
global datos
x=0;
switch evento
case 'pot1'
   j=7;       % Pin 7 de la tarjeta para contador U/D
   k=1;       % Pin 1 de la tarjeta para contador INC

calculo(str2num(get(datos.val.val1,'String')),str2num(sprintf('%3.1f',get(datos.bar.b
ar1,'Value'))),j,k);
   set(datos.val.val1,'String',sprintf('%3.1f',get(datos.bar.bar1,'Value')));
   set(datos.bar.bar1,'Value',str2num(get(datos.val.val1,'String')));
case 'pot2'
   j=10;       % Pin 10 de la tarjeta para contador U/D
   k=4;       % Pin 4 de la tarjeta para contador INC

calculo(str2num(get(datos.val.val2,'String')),str2num(sprintf('%3.1f',get(datos.bar.b
ar2,'Value'))),j,k);
   set(datos.val.val2,'String',sprintf('%3.1f',get(datos.bar.bar2,'Value')));
   set(datos.bar.bar2,'Value',str2num(get(datos.val.val2,'String')));
case 'pot3'
   j=11;       % Pin 11 de la tarjeta para contador U/D
   k=15;       % Pin 15 de la tarjeta para contador INC
```

```
calculo(str2num(get(datos.val.val3,'String')),str2num(sprintf('%3.1f',get(datos.bar.b
ar3,'Value'))),j,k);
   set(datos.val.val3,'String',sprintf('%3.1f',get(datos.bar.bar3,'Value')));
   set(datos.bar.bar3,'Value',str2num(get(datos.val.val3,'String')));
case 'pot4'
   j=14;       % Pin 14 de la tarjeta para contador U/D
   k=18;       % Pin 18 de la tarjeta para contador INC

calculo(str2num(get(datos.val.val4,'String')),str2num(sprintf('%3.1f',get(datos.bar.b
ar4,'Value'))),j,k);
   set(datos.val.val4,'String',sprintf('%3.1f',get(datos.bar.bar4,'Value')));
   set(datos.bar.bar4,'Value',str2num(get(datos.val.val4,'String')));
case 'pot5'
   set(datos.val.val5,'String',sprintf('%3.1f',get(datos.bar.bar5,'Value')));
   set(datos.bar.bar5,'Value',str2num(get(datos.val.val5,'String')));
case 'pot6'
   set(datos.val.val6,'String',sprintf('%3.1f',get(datos.bar.bar6,'Value')));
   set(datos.bar.bar6,'Value',str2num(get(datos.val.val6,'String')));
end
```

CALCULO

```
% ========== Llamado a funcion CALCULO ========== %
function calculo(a,b,j,k)
global datos
c=b-a;
if c>0
   putvalue(datos.dio.line(j),1);                    % Up
   for i=0.1:0.1:2*c
      if getvalue(datos.dio.line(k))==0               % Contador (INC)
         putvalue(datos.dio.line(k),1);
```

```
            putvalue(datos.dio.line(k+2),1);
        else
            putvalue(datos.dio.line(k),0);
            putvalue(datos.dio.line(k+2),0);
        end
    end
elseif c==0
else
    putvalue(datos.dio.line(j),0);                    % Down
    c=-c;
    for i=0.1:0.1:2*c
        if getvalue(datos.dio.line(k))==0              % Contador (INC)
            putvalue(datos.dio.line(k),1);
            putvalue(datos.dio.line(k+2),1);
        else
            putvalue(datos.dio.line(k),0);
            putvalue(datos.dio.line(k+2),0);
        end
    end
end
```

DETENER

```
% ========== Llamado a funcion DETENER ========== %
function Detener
global datos
if datos.det==0
    stop([datos.ai datos.ao]);
    set(datos.detener,'Label','Continuar Adquisicion');
    set(datos.pantalla.detener,'String','Continuar Adquisicion');
    datos.det=1;
else
```

```
   start([datos.ai datos.ao]);
   %trigger([datos.ai datos.ao]);
   set(datos.detener,'Label','Detener');
   set(datos.pantalla.detener,'String','Detener Adquisicion');
   datos.det=0;
end
```

DOMINIO

```
% ========== Llamado a funcion DOMINIO ========== %
function Dominio(evento)
global datos
switch evento
case 'dominio1'
   datos.vec(3)=get(datos.pun.pun3,'Value');
case 'dominio2'
   datos.vec(4)=get(datos.pun.pun4,'Value');
end
```

FRECUENCIA

```
% ========== Llamado a funcion Frecuencia ========== %
function [f,mag]=Frecuencia(evento)
global datos
switch evento
case 1
   if datos.vec(1)<=2
      xFFT=fft(datos.valor);
   else
      xFFT=fft(datos.canal);
   end
```

```
case 2
   if datos.vec(2)<=2
      xFFT=fft(datos.filtrar);
   else
      xFFT=fft(datos.canal);
   end
end
xfft=abs(xFFT);
indice=find(xfft==0);
xfft(indice)=1e-17;
mag=xfft;
[a,b]=size(mag);
switch evento
case 1
   if b==2
      mag=mag(1:datos.muestras,datos.vec(1));
   else
      mag=mag(1:datos.muestras,datos.vec(1)-2);
   end
case 2
   if b==2
      mag=mag(1:datos.muestras,datos.vec(2));
   else
      mag=mag(1:datos.muestras,datos.vec(2)-2);
   end
end
f=(0:length(mag)-1)*25000/datos.muestras;
f=f(:);
```

INHABILITAR

```
% ========== Llamado a funcion Inhabilitar ========== %
function Inhabilitar
global datos
if datos.cheq==0
   set(datos.bar.bar1,'Enable','Off');
   set(datos.bar.bar2,'Enable','Off');
   set(datos.bar.bar3,'Enable','Off');
   set(datos.bar.bar4,'Enable','Off');
   set(datos.bar.bar5,'Enable','Off');
   set(datos.bar.bar6,'Enable','Off');
   set(datos.pun.pun1,'Enable','Off');
   set(datos.pun.pun2,'Enable','Off');
   set(datos.pun.pun3,'Enable','Off');
   set(datos.pun.pun4,'Enable','Off');
   set(datos.inhabilita,'Label','Habilitar Controles');
   datos.cheq=1;
else
   set(datos.bar.bar1,'Enable','On');
   set(datos.bar.bar2,'Enable','On');
   set(datos.bar.bar3,'Enable','On');
   set(datos.bar.bar4,'Enable','On');
   set(datos.bar.bar5,'Enable','On');
   set(datos.bar.bar6,'Enable','On');
   set(datos.pun.pun1,'Enable','On');
   set(datos.pun.pun2,'Enable','On');
   set(datos.pun.pun3,'Enable','On');
   set(datos.pun.pun4,'Enable','On');
   set(datos.inhabilita,'Label','Inhabilitar Controles');
   datos.cheq=0;
end
```

SALIDA

```
% ========== Llamado a funcion SALIDA ========== %
function Salida(evento)
global datos
switch evento
case 'salida1'
   datos.vec(1)=get(datos.pun.pun1,'Value');
case 'salida2'
   datos.vec(2)=get(datos.pun.pun2,'Value');
end
```

IMPEDANCIA

```
% ========== Llamado a funcion IMPEDANCIA ========== %
function impedancia
global datos
imped=mean(datos.tiempo(:,8));
putvalue(datos.dio.line(22),0);  %Señal a adquirir de 5.12 V
datos.imped=abs(1000*imped/(5.04-imped));
r1=num2str(datos.imped);
set(datos.potencia.res,'String',strcat(r1,' (ohm)'));
```

POTENCIA

```
% ========== Llamado a funcion POTENCIA ========== %
function potencia
global datos
% El valor 70 es valor de ganancia del amplificador.
Vx=38*mean(datos.filtrar(:,1));
```

```
if datos.imped~=0
   P=Vx*Vx/datos.imped
else
   P=0;
end
P1=num2str(P);
set(datos.potencia.pot,'String',strcat(P1,' (vatios)'));
```

ECUALIZADOR

```
% ========== Llamado a funcion ECUALIZADOR ========== %
function Ecualizador(evento)
global datos
switch evento
case 'eq1'
   d=round(get(datos.barra.barra1,'Value'));
   if d==3
      set(datos.texto.texto1,'String','+6 dB');
      datos.barr1=3.981;
   elseif d==2
      set(datos.texto.texto1,'String','+3 dB');
      datos.barr1=1.995;
   elseif d==1
      set(datos.texto.texto1,'String','+1 dB');
      datos.barr1=1.259;
   elseif d==0
      set(datos.texto.texto1,'String','0 dB');
      datos.barr1=1;
   elseif d==-1
      set(datos.texto.texto1,'String','-1 dB');
      datos.barr1=0.794;
   elseif d==-2
```

```
        set(datos.texto.texto1,'String','-3 dB');
        datos.barr1=0.501;
    elseif d==-3
        set(datos.texto.texto1,'String','-inf dB');
        datos.barr1=0.0000000001;
    end
    set(datos.barra.barra1,'Value',d);
case 'eq2'
    d=round(get(datos.barra.barra2,'Value'));
    if d==3
        set(datos.texto.texto2,'String','+6 dB');
        datos.barr1=3.981;
    elseif d==2
        set(datos.texto.texto2,'String','+3 dB');
        datos.barr1=1.995;
    elseif d==1
        set(datos.texto.texto2,'String','+1 dB');
        datos.barr1=1.259;
    elseif d==0
        set(datos.texto.texto2,'String','0 dB');
        datos.barr1=1;
    elseif d==-1
        set(datos.texto.texto2,'String','-1 dB');
        datos.barr1=0.794;
    elseif d==-2
        set(datos.texto.texto2,'String','-3 dB');
        datos.barr1=0.501;
    elseif d==-3
        set(datos.texto.texto2,'String','-inf dB');
        datos.barr1=0.0000000001;
    end
    set(datos.barra.barra2,'Value',d);
case 'eq3'
```

```
   d=round(get(datos.barra.barra3,'Value'));
   if d==3
      set(datos.texto.texto3,'String','+6 dB');
      datos.barr1=3.981;
   elseif d==2
      set(datos.texto.texto3,'String','+3 dB');
      datos.barr1=1.995;
   elseif d==1
      set(datos.texto.texto3,'String','+1 dB');
      datos.barr1=1.259;
   elseif d==0
      set(datos.texto.texto3,'String','0 dB');
      datos.barr1=1;
   elseif d==-1
      set(datos.texto.texto3,'String','-1 dB');
      datos.barr1=0.794;
   elseif d==-2
      set(datos.texto.texto3,'String','-3 dB');
      datos.barr1=0.501;
   elseif d==-3
      set(datos.texto.texto3,'String','-inf dB');
      datos.barr1=0.0000000001;
   end
   set(datos.barra.barra3,'Value',d);
end
```

AYUDA

```
% ========== Llamado a funcion AYUDA ========== %
function Ayuda(Evento)
switch Evento
case 'AyudaA'
```

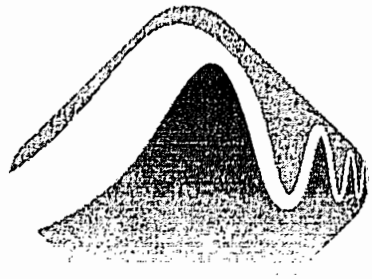
```
    open('AyudaPrincipal.htm')
case 'AyudaC'
    open('AyudaConceptos.htm')
case 'AyudaF'
    open('AyudaFuncionamiento.htm')
case 'AyudaB'
    open('AyudaCodigo.htm')
case 'Ayudal'
    open('AyudaInformacion.htm')
end
```

ANEXO C:
HOJAS DE DATOS

PCI-DAS1000, PCI-DAS1001 & PCI-DAS1002

Multifunction Analog & Digital I/O

User's Manual



MEASUREMENT COMPUTING™

Revision 2, March, 2002

© Copyright 2002, Measurement Computing Corporation

MEGA-FIFO, the CIO prefix to data acquisition board model numbers, the PCM prefix to data acquisition board model numbers, PCM-DAS08, PCM-D24C3, PCM-DAC02, PCM-COM422, PCM-COM485, PCM-DMM, PCM-DAS16D/12, PCM-DAS16S/12, PCM-DAS16D/16, PCM-DAS16S/16, PCI-DAS6402/16, Universal Library, *Insta*Cal, *Harsh Environment Warranty* and Measurement Computing Corporation are registered trademarks of Measurement Computing Corporation.

IBM, PC, and PC/AT are trademarks of International Business Machines Corp. Windows is a trademark of Microsoft Corp. All other trademarks are the property of their respective owners.

Information furnished by Measurement Computing Corp. is believed to be accurate and reliable. However, no responsibility is assumed by Measurement Computing Corporation neither for its use; nor for any infringements of patents or other rights of third parties, which may result from its use. No license is granted by implication or otherwise under any patent or copyrights of Measurement Computing Corporation.

All rights reserved. No part of this publication may be reproduced, stored in a retrieval system, or transmitted, in any form by any means, electronic, mechanical, by photocopying, recording or otherwise without the prior written permission of Measurement Computing Corporation.

NOTICE

Measurement Computing Corporation does not authorize any Measurement Computing Corporation product for use in life support systems and/or devices without the written approval of the President of Measurement Computing Corporation Life support devices/systems are devices or systems which, a) are intended for surgical implantation into the body, or b) support or sustain life and whose failure to perform can be reasonably expected to result in injury. Measurement Computing Corp. products are not designed with the components required, and are not subject to the testing required to ensure a level of reliability suitable for the treatment and diagnosis of people.

Table of Contents

1 Introduction 1-1

2 Installation 2-1

2.1 Software Installation 2-1
2.2 Hardware Installation 2-1
2.3 Connector Pinout Diagrams 2-2
2.4 Connecting Signals to the PCI-DAS1000 2-7

3 Programming & Applications 3-1

3.1 Programming Languages 3-1
3.2 Packaged Application Programs 3-1
3.3 Register Level Programming 3-1

4 Calibration 4-2

4.1 Calibration Configuration - Analog Inputs 4-2
4.2 Calibration Configuration - Analog Outputs (PCI-DAS1001 & PCI-DAS1002 Only) 4-3

5 Specifications: PCI-DAS1000 & 1001 5-1

6 Specifications: PCI-DAS1002 6-1

This page is blank.

1 Introduction

The PCI-DAS1000, PCI-DAS1001 and PCI-DAS1002 are multifunction analog and digital I/O boards designed to operate in computers with PCI bus accessory slots. The boards have the following capabilities:

- 16 single-ended or eight differential analog inputs with sample rates as high as 250 kHz.
- 24-bits of parallel digital I/O
- Three, user-accessible, 16-bit counters.
- Two analog output channels (PCI-DAS1001 and PCI-DAS1002 only)

The board's analog input ranges are as follows:

PCI-DAS1000, PCI-DAS1002	Bipolar:	±10V, ±5V, ±2.5V, and ±1.25V
	Unipolar:	0 to 10V, 0 to 5V, 0 to 2.5V and 0 to 1.25V
PCI-DAS1001	Bipolar:	±10V, ±1.0V, ±0.1V, ±0.01V
	Unipolar:	0 to 10V, 0 to 1.0V, 0 to 0.1V, 0 to 0.01V

The PCI-DAS1000 series is fully plug-and-play with no switches or jumpers to set. The boards are self-calibrating with no potentiometers to adjust. All calibration is performed via software and on-board trim D/A converters.

The PCI-DAS1000 series is fully supported by the powerful Universal Library software as well as a wide variety of application software packages including SoftWIRE.

NOTE:

Unless a specific model code is required, this manual references the PCI-DAS1000 as a general term.

2 Installation

2.1 Software Installation

The board has no switches or jumpers to set. The simplest way to configure your board is to use the InstaCal™ program provided on the CD (or floppy disk). InstaCal™ will create a configuration file that your application software (and the optional Universal Library™) will refer to so the software you use will automatically have access to the exact configuration of the board.

Please refer to the *Software Installation Manual* regarding the installation and operation of InstaCal.

2.2 Hardware Installation

The PCI-DAS1000 series boards are completely plug and play. There are no switches or jumpers to set. Configuration is controlled by your systems' BIOS. Follow the steps shown below to install your PCI board.

WARNING

Do not unplug the computer when installing the board.
Doing so removes the computer's ground.

1. Turn your computer off, open it up, and insert the PCI-DAS1000 board into any available PCI slot.
2. Close your computer up and turn it on.
3. If you are using an operating system with support for Plug and Play (such as Windows 95 or 98), a dialog box will pop up as the system loads indicating that new hardware has been detected. If the information file for this board is not already loaded onto your PC, you will be prompted for a disk containing it. The InstaCal software supplied with your board contains this file. Insert the disk or CD and click OK.

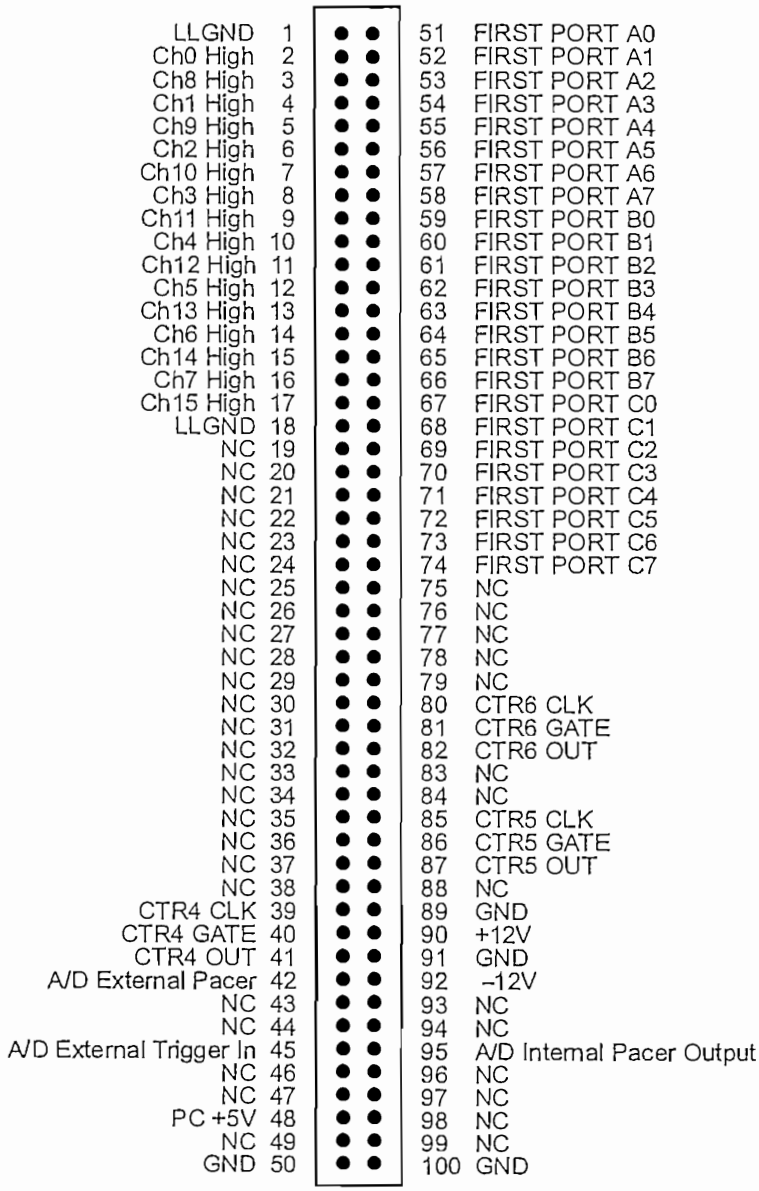
2.3 Connector Pinout Diagrams

The PCI-DAS1000 series boards use a 100-pin I/O connector. See Figures 2-1 and 2-2 for the PCI-DAS1000.
See Figures 2-3 and 2-4 for the PCI-DAS1001 and PCI-DAS1002, 8-channel differential and 16-channel single ended respectively.

LLGND	1	51	FIRST PORT A0
CH0 HI	2	52	FIRST PORT A1
CH0 LO	3	53	FIRST PORT A2
CH1 HI	4	54	FIRST PORT A3
CH1 LO	5	55	FIRST PORT A4
CH2 HI	6	56	FIRST PORT A5
Ch2 LO	7	57	FIRST PORT A6
Ch3 HI	8	58	FIRST PORT A7
Ch3 LO	9	59	FIRST PORT B0
CH4 HI	10	60	FIRST PORT B1
Ch4 LO	11	61	FIRST PORT B2
Ch5 HI	12	62	FIRST PORT B3
Ch5 LO	13	63	FIRST PORT B4
Ch6 HI	14	64	FIRST PORT B5
Ch6 LO	15	65	FIRST PORT B6
Ch7 HI	16	66	FIRST PORT B7
Ch7 LO	17	67	FIRST PORT C0
LLGND	18	68	FIRST PORT C1
NC	19	69	FIRST PORT C2
NC	20	70	FIRST PORT C3
NC	21	71	FIRST PORT C4
NC	22	72	FIRST PORT C5
NC	23	73	FIRST PORT C6
NC	24	74	FIRST PORT C7
NC	25	75	NC
NC	26	76	NC
NC	27	77	NC
NC	28	78	NC
NC	29	79	NC
NC	30	80	CTR6 CLK
NC	31	81	CTR6 GATE
NC	32	82	CTR6 OUT
NC	33	83	NC
NC	34	84	NC
NC	35	85	CTR5 CLK
NC	36	86	CTR5 GATE
NC	37	87	CTR5 OUT
NC	38	88	NC
CTR4 CLK	39	89	GND
CTR4 GATE	40	90	+12V
CTR4 OUT	41	91	GND
A/D External Pacer	42	92	-12V
NC	43	93	NC
NC	44	94	NC
A/D External Trigger In	45	95	A/D Internal Pacer Output
NC	46	96	NC
NC	47	97	NC
PC +5V	48	98	NC
NC	49	99	NC
GND	50	100	GND

PCI-DAS1000 Connector Diagram
8 Channel, DIFFERENTIAL ANALOG-IN

Figure 2-1. Connector Diagram for PCI-DAS1000 - 8-Channel Differential Usage



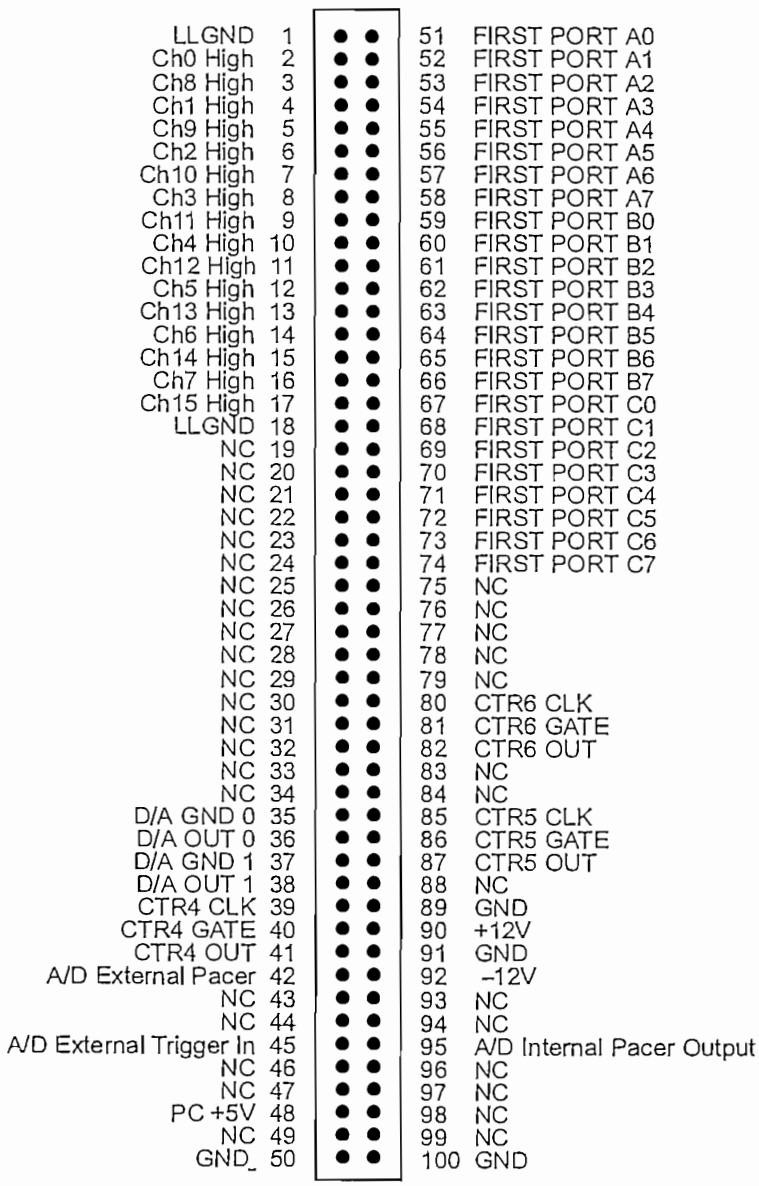
PCI-DAS1000 Connector Diagram
16 Channel, Single-Ended

Figure 2-2. Connector Diagram for PCI-DAS1000 - 16-Channel Single-Ended Usage

LLGND	1	51	FIRST PORT A0
CH0 HI	2	52	FIRST PORT A1
CH0 LO	3	53	FIRST PORT A2
CH1 HI	4	54	FIRST PORT A3
CH1 LO	5	55	FIRST PORT A4
CH2 HI	6	56	FIRST PORT A5
Ch2 LO	7	57	FIRST PORT A6
Ch3 HI	8	58	FIRST PORT A7
Ch3 LO	9	59	FIRST PORT B0
CH4 HI	10	60	FIRST PORT B1
Ch4 LO	11	61	FIRST PORT B2
Ch5 HI	12	62	FIRST PORT B3
Ch5 LO	13	63	FIRST PORT B4
Ch6 HI	14	64	FIRST PORT B5
Ch6 LO	15	65	FIRST PORT B6
Ch7 HI	16	66	FIRST PORT B7
Ch7 LO	17	67	FIRST PORT C0
LLGND	18	68	FIRST PORT C1
NC	19	69	FIRST PORT C2
NC	20	70	FIRST PORT C3
NC	21	71	FIRST PORT C4
NC	22	72	FIRST PORT C5
NC	23	73	FIRST PORT C6
NC	24	74	FIRST PORT C7
NC	25	75	NC
NC	26	76	NC
NC	27	77	NC
NC	28	78	NC
NC	29	79	NC
NC	30	80	CTR6 CLK
NC	31	81	CTR6 GATE
NC	32	82	CTR6 OUT
NC	33	83	NC
NC	34	84	NC
D/A GND 0	35	85	CTR5 CLK
D/A OUT 0	36	86	CTR5 GATE
D/A GND 1	37	87	CTR5 OUT
D/A OUT 1	38	88	NC
CTR4 CLK	39	89	GND
CTR4 GATE	40	90	+12V
CTR4 OUT	41	91	GND
A/D External Pacer	42	92	–12V
NC	43	93	NC
NC	44	94	NC
A/D External Trigger In	45	95	A/D Internal Pacer Output
NC	46	96	NC
NC	47	97	NC
PC +5V	48	98	NC
NC	49	99	NC
GND	50	100	GND

PCI-DAS1001 & PCI-DAS1002 Connector Diagram
8 Channel, DIFFERENTIAL ANALOG-IN

Figure 2-3. Connector Diagram for PCI-DAS1001 & 1002 - 8-Channel Differential Usage



PCI-DAS1001 & PCI-DAS1002 Connector Diagram
16 Channel, Single-Ended

Figure 2-4. Connector Diagram for PCI-DAS1001 & 1002 - 16-Channel Single-EndedUsage

2.4 Connecting Signals to the PCI-DAS1000

The 100-pin connector provides a far greater signal density than the traditional 37-pin D-type connector. For a mating cable, use the C100FF-2. This cable assembly has a 100-pin connector that fans out to a pair of 50-pin ribbon cables. The two 50-pin ribbon cable legs are terminated with standard 50-pin header connectors. A pair of CIO-MINI50 screw terminal boards, a single CIO-TERM100 screw terminal board or a single SCB-50 breakout box can be used to terminate field signals and route them to the PCI-DAS1000. The BNC16/8 series interface box provides convenient and reliable BNC connections to each of the analog outputs.

There is additional information regarding analog signal connection and configuration at http://www.measurementcomputing.com/signals/signals.pdf.

3 Programming & Applications

3.1 Programming Languages

The Universal Library provides complete access to the PCI-DAS1000 functions from the full range of Windows programming languages. If you are planning to write programs, or would like to run the example programs for Visual Basic or any other language, please refer to the Universal Library manual.

3.2 Packaged Application Programs

Many packaged application programs, such as SoftWIRE have drivers for the PCI-DAS1000. If the software you use does not have drivers for the PCI-DAS1000, please fax or e-mail the package name and the revision number from the install disks. We will research the package for you and advise how to obtain the necessary drivers.

Some application drivers are included with the Universal Library package, but not with the Application package. If you have purchased an application package directly from the software vendor, you may need to purchase our Universal Library and drivers. Please contact us for more information on this topic.

3.3 Register Level Programming

The PCI-DAS1000 is supported by the powerful Universal Library. We strongly recommend that you take advantage of the Universal Library as your software interface. The complexity of registers used for automatic calibration, combined with the PCI BIOS's dynamic allocation of addresses and internal resources, make the PCI-DAS1000 series challenging to program via direct register I/O operations. Direct I/O programming should be attempted only by experienced programmers.

Although the PCI-DAS1000 is part of the larger DAS family, there is no correspondence between register locations of the PCI-DAS1000 and boards in the CIO-DAS16 family. Software written at the register level for the other DAS boards will not work with the PCI-DAS1000. However, software based on the Universal Library should work with the PCI-DAS1000 with few or no changes.

If you decide that register level programming is required for your application, information on the register functions can be found at
http://www.measurementcomputing.com/registermaps/.

4 Calibration

The PCI-DAS1000 is shipped fully-calibrated from the factory with calibration coefficients stored in nvRAM. At run time, these calibration factors are loaded into system memory and are automatically retrieved each time a different DAC/ADC range is specified. The user has the option to recalibrate with respect to the factory-measured voltage standards at any time by selecting the "Calibrate" option in InstaCal. Full calibration typically requires less than two minutes and requires no other user intervention.

4.1 Calibration Configuration - Analog Inputs

The PCI-DAS1000 provides self-calibration for the analog input and measurement circuits, eliminating the need for external equipment and user adjustments. All adjustments are made via 8-bit calibration DACs or 7-bit digital potentiometers referenced to an on-board factory-calibrated standard. Calibration factors are stored on the serial nvRAM.

A variety of methods are used to calibrate the different elements on the board. The analog front-end has several software "knobs" to turn. Offset calibration is performed in the instrumentation amplifier gain stage. Front-end gain adjustment is performed via a variable attenuator/gain stage.

Figure 4-1 is a block diagram of the analog input front-end calibration system:

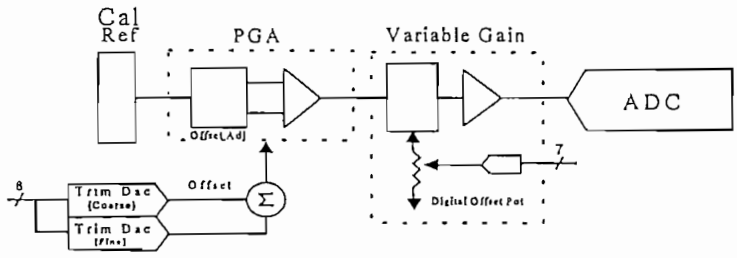


Figure4-1. Analog Front-End Calibration Block Diagram

4.2 Calibration Configuration - Analog Outputs (PCI-DAS1001 & PCI-DAS1002 Only)

The calibration scheme for analog outputs is in Figure 4-2. The function is duplicated for DAC0 and DAC1.

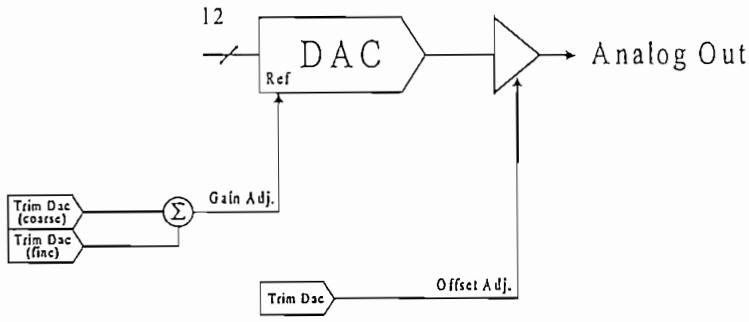


Figure 4-2. Analog Output Calibration Block Diagram - PCI-DAS1001 and PCI-DAS1002 Only

5 Specifications: PCI-DAS1000 & 1001

Typical for 25°C unless otherwise specified

Power Consumption

+5V Operating (A/D converting to FIFO)	0.8A typical, 1.0A max

Analog input Section

A/D converter type	7800
Resolution	12 bits
Number of channels	8 differential or 16 single-ended, software selectable
Input Ranges PCI-DAS1000 PCI-DAS1001	±10V, ±5V, ±2.5V, ±1.25V, 0 to 10V, 0 to 5V, 0 to 2.5V, 0 to 1.25V fully programmable ±10V, ±1V, ±0.1V, ±0.01V, 0 to 10V, 0 to 1V, 0 to 0.1V, 0 to 0.01V fully programmable
Polarity	Unipolar/Bipolar, software selectable
A/D pacing	Programmable: internal counter or external source (A/D External Pacer, positive or negative edge selectable by software) or software polled
Burstmode PCI-DAS1000 PCI-DAS1001	 Software selectable option, rate = 4µs Software selectable option, rate = 6.67µs
A/D Trigger sources	External digital (A/D External Trigger)
A/D Triggering Modes Digital: Pre-trigger: Data transfer: A/D conversion time:	 Software enabled, rising edge, hardware trigger Unlimited pre- and post-trigger samples. Total # of samples must be > 512. From 1024-sample FIFO via REPINSW, interrupt, or software polled 3 µs

Throughput PCI-DAS1001 PCI-DAS1000	 150 kHz 250 kHz
Relative Accuracy	±1.5 LSB
Differential Linearity error:	±0.75 LSB
Integral Linearity error	±0.5 LSB typ, ±1.5 LSB max
Gain Error (relative to calibration reference) 0.01V Range All other Ranges	 ± 0.4% of reading Max ± 0.02% of reading Max
No missing codes guaranteed	12 bits
Calibration	Auto-calibration, calibration factors for each range stored on board in non-volatile RAM
Gain drift (A/D specs)	±6ppm/°C
Zero drift (A/D specs)	±1ppm/°C
Common Mode Range	±10V
CMRR @ 60Hz	70dB
Input leakage current	200nA
Input impedance	10Meg Ohms Min
Absolute maximum input voltage PCI-DAS1001 PCI-DAS1000 Noise Distribution (Rate = 1-250KHz, Average % ±2 bins, Average % ±1 bin, Average # bins) PCI-DAS1000 All Bipolar ranges All Unipolar ranges PCI-DAS1001 10V Ranges 1V Ranges 0.1V Ranges Bipolar 0.01V Range Unipolar 0.01V Range	 ±35V Channels 1-15: -40V to +55V power on or off Channel 0 : ±15V 100% / 99.5% / 4 bins 100% / 99% / 5 bins 3 bins (100%) 4 bins (100%) 10 bins (100%) 20 bins (100%) 32 bins (100%)

Analog Output (PCI-DAS1001 only)

D/A type	AD7847AR
Resolution	12 bits
Number of channels:	2
Output Ranges	±10V, ±5V, 0 to 5V, 0 to 10V. Each channel independently programmable.

D/A pacing	Software
Data transfer	Programmed I/O.
Offset error	±600µV max, all ranges (calibrated)
Gain error	±0.02% FSR max (calibrated)
Differential nonlinearity	±1LSB max
Integral nonlinearity	±1LSB max
Monotonicity	12 bits
D/A Gain drift	±2 ppm/°C max
D/A Bipolar offset drift	±5 ppm/°C max
D/A Unipolar offset drift	±5 ppm/°C max

Throughput	PC-dependent
Settling time (to 0.01% of 10V step):	4µs typ
Slew Rate	7V/µS

Current Drive	±5 mA min
Output short-circuit duration	25 mA indefinite
Output Coupling	DC
Amp Output Impedance	0.1 Ohms max

Miscellaneous	Power up and reset, all DAC's cleared to 0 volts, ±200mV

Digital Input / Output

Digital Type	82C55A
Configuration	2 banks of 8, 2 banks of 4, programmable by bank as input or output
Number of channels	24 I/O
Output High	3.0 volts min @ –2.5mA
Output Low	0.4 volts max @ 2.5 mA
Input High	2.0 volts min, +5.5 volts absolute max
Input Low	0.8 volts max, -0.5 volts absolute min
Power-up / reset state	Input mode (high impedance)
Interrupts	INTA# - mapped to IRQn via PCI BIOS at boot-time
Interrupt enable	Programmable
Interrupt sources	Residual counter, End-of-channel-scan, AD-FIFO-not-empty, AD-FIFO-half-full

Counter section

Counter type	82C54
Configuration	Two 82C54 devices. 3 down counters per 82C54, 16 bits each
82C54A:	Counter 0: ADC residual sample counter. Source: ADC Clock Gate: Internal programmable source Output: End-of-Acquisition interrupt Counter 1: ADC Pacer Lower Divider Source: 10 MHz oscillator Gate: Tied to Counter 2 gate, programmable source. Output: Chained to Counter 2 Clock. Counter 2: ADC Pacer Upper Divider Source: Counter 1 Output. Gate: Tied to Counter 1 gate, programmable source. Output: ADC Pacer clock (if software selected), available at user connector (A/D Internal Pacer Output).

82C54B:	Counter 0: Pretrigger Mode Source: ADC Clock. Gate: External trigger Output: End-of-Acquisition interrupt Counter 0 - User Counter 4 (when in non-Pretrigger Mode) Source: User input at 100pin connector (CTR4 CLK) or internal 10MHz (software selectable) Gate: User input at 100pin connector (CTR4 GATE) Output: Available at 100pin connector (CTR4 OUT) Counter 1: User Counter 5 Source: User input at 100pin connector (CTR5 CLK) Gate: User input at 100pin connector (CTR5 GATE) Output: Available at 100pin connector (CTR5 OUT) Counter 2: User Counter 6 Source: User input at 100pin connector (CTR6 CLK) Gate: User input at 100pin connector (CTR6 GATE) Output: Available at 100pin connector (CTR6 OUT)
Clock input frequency	10 Mhz max
High pulse width (clock input)	30 ns min
Low pulse width (clock input)	50 ns min
Gate width high	50 ns min
Gate width low	50 ns min
Input low voltage	0.8V max
Input high voltage	2.0V min
Output low voltage	0.4V max
Output high voltage	3.0V min

Environmental

Operating temperature range	0 to 70°C
Storage temperature range	–40 to 100°C
Humidity	0 to 90% non-condensing

6 Specifications: PCI-DAS1002

Typical for 25°C unless otherwise specified.

Power consumption

+5V	0.8A typical, 1.0A max

Analog input section

A/D converter type	ADS7800 or equivalent
Resolution	12 bits
Number of channels	16 single-ended / 8 differential, software selectable
Input ranges	±10V, ±5V, ±2.5V, ±1.25V, 0 to 10V, 0 to 5V, 0 to 2.5V, 0 to 1.25V software programmable
A/D pacing (software programmable)	Internal counter - 82C54.
	External source (A/D External Pacer) software programmable for rising or falling edge
	Software polled
A/D trigger sources	External edge trigger (A/D External Trigger)
A/D triggering modes	Rising or falling edge trigger - software selectable
A/D pre-trigger mode	Unlimited pre- and post-trigger samples. Total number of samples must be greater than 512.
Burst mode	Software selectable option, burst rate = 150kHz
Data transfer	From 1024 sample FIFO via REPINSW
	Programmed I/O
A/D conversion time	3µs max
Analog front end settling time	6µs for a full scale step to 1 LSB
Calibrated throughput	200KHz
Calibration	Auto-calibration, calibration factors for each range stored on board in nonvolatile RAM

Accuracy

Accuracies are listed for a 200KHz sampling rate, 100 sample average, single channel operation, a 15 minute warm-up, and operational temperatures within ±2degC of internal calibration temperature. The calibrator test source high side is tied to Channel 0 In and the low side tied to AGND.

Table 1 – *Absolute Accuracy*

Range	Absolute Accuracy
±10.00V	±2.5 LSB max
±5.000V	±2.5 LSB max
±2.500V	±2.5 LSB max
±1.250V	±2.5 LSB max
0 to 10.00V	±2.5 LSB max
0 to 5.000V	±2.5 LSB max
0 to 2.500V	±2.5 LSB max
0 to 1.250V	±2.5 LSB max

Each PCI-DAS1002 is tested at the factory to assure the board's overall error does not exceed accuracy limits described in *Table 1* above.

Table 2 – *Calibrated Accuracy Components (in LSB)*

Range	Gain Error	Offset Error	DLE	ILE
All ranges	±1.0 max	±1.0 max	±0.75 max	±0.5 max

As shown in *Table 2*, total board error is a combination of Gain, Offset, Differential Linearity and Integral Linearity error. The theoretical worst-case error of the board may be calculated by summing these component errors. Worst case errors are realized only in the unlikely event that each of the component errors are at their maximum level, and causing error in the same direction.

Crosstalk

Crosstalk is defined here as the influence of one channel upon another when scanning two channels at the maximum rate. A full scale 100Hz triangle wave is input on Channel 1; Channel 0 is tied to Analog Ground at the 100 pin user connector. The table below summarizes the influence of Channel 1 on Channel 0 with the effects of noise removed. The residue on Channel zero is described in LSB's.

Condition	Crosstalk	Per channel Rate	ADC Rate
All Ranges	2 LSB $_{pk-pk}$	100 KHz	200 KHz

A/D Full-Scale Gain drift	±0.25 LSB/°C max
A/D Zero drift	±0.25 LSB/°C max
Common Mode Range	±10V min
CMRR @ 60Hz	-70dB min
Input leakage current	±20nA max
Input impedance	10 MOhms min
Absolute maximum input voltage	±35 volts
Warm-up time	15 minutes

Noise Performance

Table 3 below summarizes the noise performance for the PCI-DAS1002. Noise distribution is determined by gathering 50K samples at 200kHz with inputs tied to ground at the user connector.

Table 3 – *Board Noise Performance*

Range	% within ±2 LSB	% within ±1 LSB	LSBs	LSBrms*
0 to 1.250V	100%	99%	4	0.61
All other ranges	100%	100%	3	0.45

* RMS noise is defined as the peak-to-peak bin spread divided by 6.6

Analog output section

D/A converter type	AD7847AR or equivalent
Resolution	12 bits
Number of channels	2
Configuration	Voltage Output, Single-ended
Output Range	±10V, ±5V, 0 to 10V, or 0 to 5V. Software selectable. Each channel independently programmable.
D/A pacing	Software
Data transfer	Programmed I/O

Absolute Accuracy

All Ranges	±3 LSB

Calibrated Accuracy Components

Gain Error	±1.0 LSB max
Offset Error	±0.5LSB max
Integral Linearity Error	±1.0 LSB max
Differential Linearity Error	±1.0 LSB max

Each PCI-DAS1002 is tested at the factory to assure absolute accuracy.

Total board error is a combination of Gain, Offset, Integral Linearity and Differential Linearity error. The theoretical worst-case error of the board may be calculated by summing these component errors. Worst case error is realized only in the unlikely event that each of the component errors are at their maximum level, and causing error in the same direction. Although an examination of the chart and a summation of the maximum theoretical errors shows that the board could theoretically exhibit a ±3.5 LSB error, our testing assures this error is never realized in a board that we ship.

Monotonicity	Guaranteed monotonic over temperature
Overall Analog Output drift	±0.03 LSB/°C max
Settling time	4µs to 0.01% of 10V step
Slew Rate	7V/µs min
Current Drive	±5 mA min
Output short-circuit duration	Indefinite @25mA
Output coupling	DC
Output impedance	0.1 ohms max
Miscellaneous	Double buffered output latches
	Output voltage on power up and reset: ±200mV

Counter section

Counter type	82C54	
Configuration	Two 82C54 devices, 3 down counters per 82C54, 16 bits each	
Counter 1 – ADC residual sample counter	Source:	ADC clock
	Gate:	Internal programmable source
	Output:	End of acquisition interrupt
Counter 2 - ADC Pacer Lower Divider	Source:	10 MHz internal source
	Gate:	Internal, programmable on/off
	Output:	Chained to Counter 3 Clock
Counter 3 - ADC Pacer Upper Divider	Source:	Counter 2 Output
	Gate:	Internal, programmable on/off
	Output:	Programmable as ADC Pacer clock. Available at user connector (ADC Pacer out)
Counter 4 – Pre-trigger mode	Source:	ADC clock for pre-trigger mode
	Gate:	External trigger for pre-trigger mode
	Output:	End of acquisition interrupt for pre-trigger mode
Counter 4 – Non-Pre-Trigger mode	Source:	External at connector (CTR4 CLK)
	Gate:	External at connector (CTR4 GATE)
	Output:	Available at connector (CTR4 OUT)
Counter 5 - User counter	Source:	External at connector (CTR5 CLK)
	Gate:	External at connector (CTR5 GATE)
	Output:	Available at connector (CTR5 OUT)
Counter 6 - User counter	Source:	External at connector (CTR6 CLK)
	Gate:	External at connector (CTR6 GATE)
	Output:	Available at connector (CTR6 OUT)
Clock input frequency	10Mhz max	
High pulse width (clock input)	30ns min	
Low pulse width (clock input)	50ns min	
Gate width high	50ns min	
Gate width low	50ns min	
Input low voltage	0.8V max	
Input high voltage	2.0V min	
Output low voltage	0.4V max	

Output high voltage	3.0V min
Crystal Oscillator Frequency	10MHz
Frequency accuracy	50ppm

Digital input/output section

Digital Type	82C55
Number of I/O	24 (Port A0 through Port C7)
Configuration	• 2 banks of 8 and 2 banks of 4 or
	• 3 banks of 8 or
	• 2 banks of 8 with handshake
Input high voltage	2.0V min, 5.5V absolute max
Input low voltage	0.8V max, –0.5V absolute min
Output high voltage (IOH = -2.5mA)	3.0V min
Output low voltage (IOL = 2.5mA)	0.4V max
Power-up / reset state	Input mode (high impedance)

Interrupt section

Interrupts	PCI INTA# - mapped to IRQn via PCI BIOS at boot-time
Interrupt enable	Programmable. Default = disabled.
Interrupt sources	• Residual sample counter • A/D End-of-channel-scan • A/D FIFO-not-empty • A/D FIFO-half-full • A/D Pacer

Miscellaneous

+5 Volts	Available at I\O connector (PC+5V)
+12 Volts	Available at I\O connector (PC +12V)
-12 Volts	Available at I\O connector (PC –12V)

Environmental

Operating Temperature Range	0 to 70°C
Storage Temperature Range	-40 to 100°C
Humidity	0 to 95% non-condensing

Mechanical

Card dimensions	PCI half card: 174.63mm(L) x 106.86mm(H) x 14.48mm(D)

Connector and Pin Out

8 Channel Differential Mode

Pin	Signal Name	Pin	Signal Name
1	LLGND	51	FIRST PORT A 0
2	CH0 HI	52	FIRST PORT A 1
3	CH0 LO	53	FIRST PORT A 2
4	CH1 HI	54	FIRST PORT A 3
5	CH1 LO	55	FIRST PORT A 4
6	CH2 HI	56	FIRST PORT A 5
7	CH2 LO	57	FIRST PORT A 6
8	CH3 HI	58	FIRST PORT A 7
9	CH3 LO	59	FIRST PORT B 0
10	CH4 HI	60	FIRST PORT B 1
11	CH4 LO	61	FIRST PORT B 2
12	CH5 HI	62	FIRST PORT B 3
13	CH5 LO	63	FIRST PORT B 4
14	CH6 HI	64	FIRST PORT B 5
15	CH6 LO	65	FIRST PORT B 6
16	CH7 HI	66	FIRST PORT B 7
17	CH7 LO	67	FIRST PORT C 0
18	LLGND	68	FIRST PORT C 1
19	N/C	69	FIRST PORT C 2
20	N/C	70	FIRST PORT C 3
21	N/C	71	FIRST PORT C 4
22	N/C	72	FIRST PORT C 5
23	N/C	73	FIRST PORT C 6
24	N/C	74	FIRST PORT C 7
25	N/C	75	N/C
26	N/C	76	N/C
27	N/C	77	N/C
28	N/C	78	N/C
29	N/C	79	N/C
30	N/C	80	CTR6 CLK
31	N/C	81	CTR6 GATE
32	N/C	82	CTR6 OUT
33	N/C	83	N/C
34	N/C	84	N/C
35	D/A GND 0	85	CTR5 CLK
36	D/A OUT 0	86	CTR5 GATE
37	D/A GND 1	87	CTR5 OUT
38	D/A OUT 1	88	N/C
39	CTR4 CLK	89	GND
40	CTR4 GATE	90	+12V
41	CTR4 OUT	91	GND
42	A/D EXTERNAL PACER	92	-12V
43	N/C	93	N/C
44	N/C	94	N/C
45	A/D EXTERNAL TRIGGER IN	95	A/D INTERNAL PACER OUTPUT
46	N/C	96	N/C
47	N/C	97	N/C
48	PC +5V	98	N/C
49	N/C	99	N/C
50	GND	100	GND

16 Channel Single-Ended Mode

Pin	Signal Name	Pin	Signal Name
1	LLGND	51	FIRST PORT A 0
2	CH0 HI	52	FIRST PORT A 1
3	CH8 HI	53	FIRST PORT A 2
4	CH1 HI	54	FIRST PORT A 3
5	CH9 HI	55	FIRST PORT A 4
6	CH2 HI	56	FIRST PORT A 5
7	CH10 HI	57	FIRST PORT A 6
8	CH3 HI	58	FIRST PORT A 7
9	CH11 HI	59	FIRST PORT B 0
10	CH4 HI	60	FIRST PORT B 1
11	CH12 HI	61	FIRST PORT B 2
12	CH5 HI	62	FIRST PORT B 3
13	CH13 HI	63	FIRST PORT B 4
14	CH6 HI	64	FIRST PORT B 5
15	CH14 HI	65	FIRST PORT B 6
16	CH7 HI	66	FIRST PORT B 7
17	CH15 HI	67	FIRST PORT C 0
18	LLGND	68	FIRST PORT C 1
19	N/C	69	FIRST PORT C 2
20	N/C	70	FIRST PORT C 3
21	N/C	71	FIRST PORT C 4
22	N/C	72	FIRST PORT C 5
23	N/C	73	FIRST PORT C 6
24	N/C	74	FIRST PORT C 7
25	N/C	75	N/C
26	N/C	76	N/C
27	N/C	77	N/C
28	N/C	78	N/C
29	N/C	79	N/C
30	N/C	80	CTR6 CLK
31	N/C	81	CTR6 GATE
32	N/C	82	CTR6 OUT
33	N/C	83	N/C
34	N/C	84	N/C
35	D/A GND 0	85	CTR5 CLK
36	D/A OUT 0	86	CTR5 GATE
37	D/A GND 1	87	CTR5 OUT
38	D/A OUT 1	88	N/C
39	CTR4 CLK	89	GND
40	CTR4 GATE	90	+12V
41	CTR4 OUT	91	GND
42	A/D EXTERNAL PACER	92	-12V
43	N/C	93	N/C
44	N/C	94	N/C
45	A/D EXTERNAL TRIGGER IN	95	A/D INTERNAL PACER OUTPUT
46	N/C	96	N/C
47	N/C	97	N/C
48	PC +5V	98	N/C
49	N/C	99	N/C
50	GND	100	GND

EC Declaration of Conformity

We, Measurement Computing Corporation, declare under sole responsibility that the products:

Part Number	Description
PCI-DAS1000	High speed analog input board for the PCI bus
PCI-DAS1001	High speed analog input board for the PCI bus
PCI-DAS1002	High speed analog input board for the PCI bus

to which this declaration relates, meets the essential requirements, is in conformity with, and CE marking has been applied according to the relevant EC Directives listed below using the relevant section of the following EC standards and other normative documents:

EU EMC Directive 89/336/EEC: Essential requirements relating to electromagnetic compatibility.

EU 55022 Class B: Limits and methods of measurements of radio interference characteristics of information technology equipment.

EN 50082-1: EC generic immunity requirements.

IEC 801-2: Electrostatic discharge requirements for industrial process measurement and control equipment.

IEC 801-3: Radiated electromagnetic field requirements for industrial process measurements and control equipment.

IEC 801-4: Electrically fast transients for industrial process measurement and control equipment.

Carl Haapaoja, Director of Quality Assurance

Measurement Computing Corporation
16 Commerce Boulevard,
Middleboro, Massachusetts 02346
(508) 946-5100
Fax: (508) 946-9500
E-mail: info@measurementcomputing.com
www. measurementcomputing.com

TL081, TL081A, TL081B, TL082, TL082A, TL082B
TL082Y, TL084, TL084A, TL084B, TL084Y
JFET-INPUT OPERATIONAL AMPLIFIERS

SLOS081C – FEBRUARY 1977 – REVISED SEPTEMBER 1996

24 DEVICES COVER COMMERCIAL, INDUSTRIAL, AND MILITARY TEMPERATURE RANGES

- Low Power Consumption
- Wide Common-Mode and Differential Voltage Ranges
- Low Input Bias and Offset Currents
- Output Short-Circuit Protection
- Low Total Harmonic Distortion . . . 0.003% Typ
- High Input Impedance . . . JFET-Input Stage
- Latch-Up-Free Operation
- High Slew Rate . . . 13 V/μs Typ
- Common-Mode Input Voltage Range Includes V_{CC+}

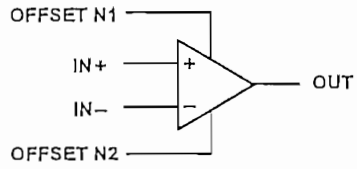
description

The TL08x JFET-input operational amplifier family is designed to offer a wider selection than any previously developed operational amplifier family. Each of these JFET-input operational amplifiers incorporates well-matched, high-voltage JFET and bipolar transistors in a monolithic integrated circuit. The devices feature high slew rates, low input bias and offset currents, and low offset voltage temperature coefficient. Offset adjustment and external compensation options are available within the TL08x family.

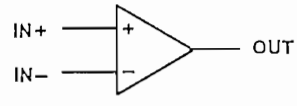
The C-suffix devices are characterized for operation from 0°C to 70°C. The I-suffix devices are characterized for operation from –40°C to 85°C. The M-suffix devices are characterized for operation over the full military temperature range of –55°C to 125°C.

symbols

TL081



TL082 (EACH AMPLIFIER)
TL084 (EACH AMPLIFIER)



Please be aware that an important notice concerning availability, standard warranty, and use in critical applications of Texas Instruments semiconductor products and disclaimers thereto appears at the end of this data sheet.

PRODUCTION DATA Information is current as of publication date. Products conform to specifications per the terms of Texas Instruments standard warranty. Production processing does not necessarily include testing of all parameters.

Copyright © 1996, Texas Instruments Incorporated

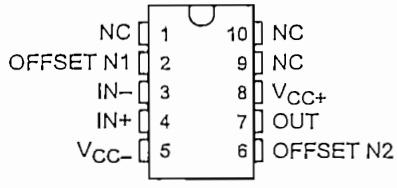
On products compliant to MIL-PRF-38535, all parameters are tested unless otherwise noted. On all other products, production processing does not necessarily include testing of all parameters.

TEXAS INSTRUMENTS
POST OFFICE BOX 655303 • DALLAS, TEXAS 75265

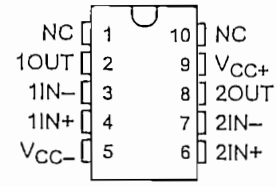
TL081, TL081A, TL081B, TL082, TL082A, TL082B
TL082Y, TL084, TL084A, TL084B, TL084Y
JFET-INPUT OPERATIONAL AMPLIFIERS

SLOS081C – FEBRUARY 1977 – REVISED SEPTEMBER 1996

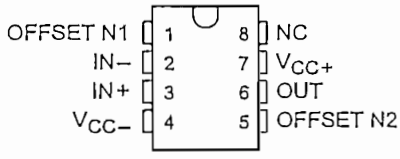
TL081, TL081A, TL081B
U PACKAGE
(TOP VIEW)



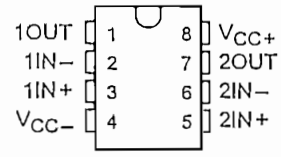
TL082, TL082A, TL082B
U PACKAGE
(TOP VIEW)



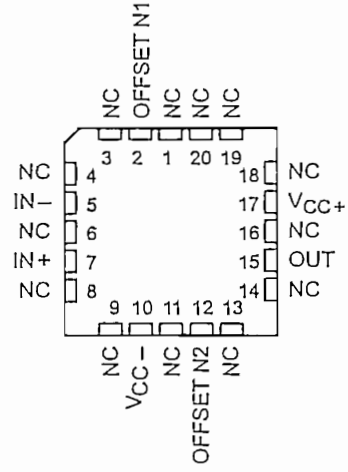
TL081, TL081A, TL081B
D, JG, P, OR PW PACKAGE
(TOP VIEW)



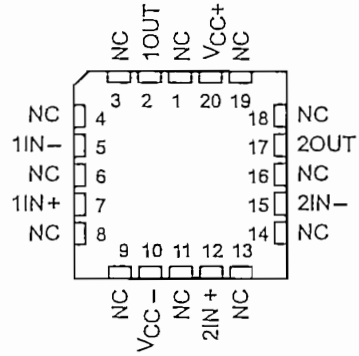
TL082, TL082A, TL082B
D, JG, P, OR PW PACKAGE
(TOP VIEW)



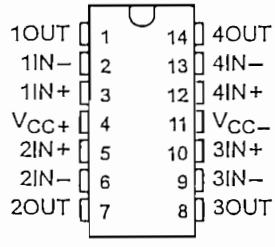
TL081M . . . FK PACKAGE
(TOP VIEW)



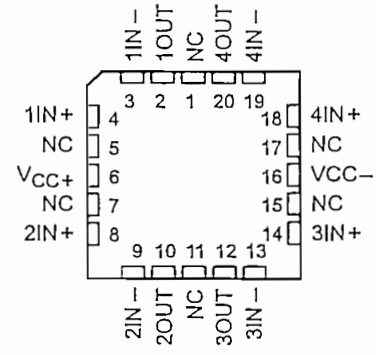
TL082M . . . FK PACKAGE
(TOP VIEW)



TL084, TL084A, TL084B
D, J, N, PW, OR W PACKAGE
(TOP VIEW)



TL084M . . . FK PACKAGE
(TOP VIEW)



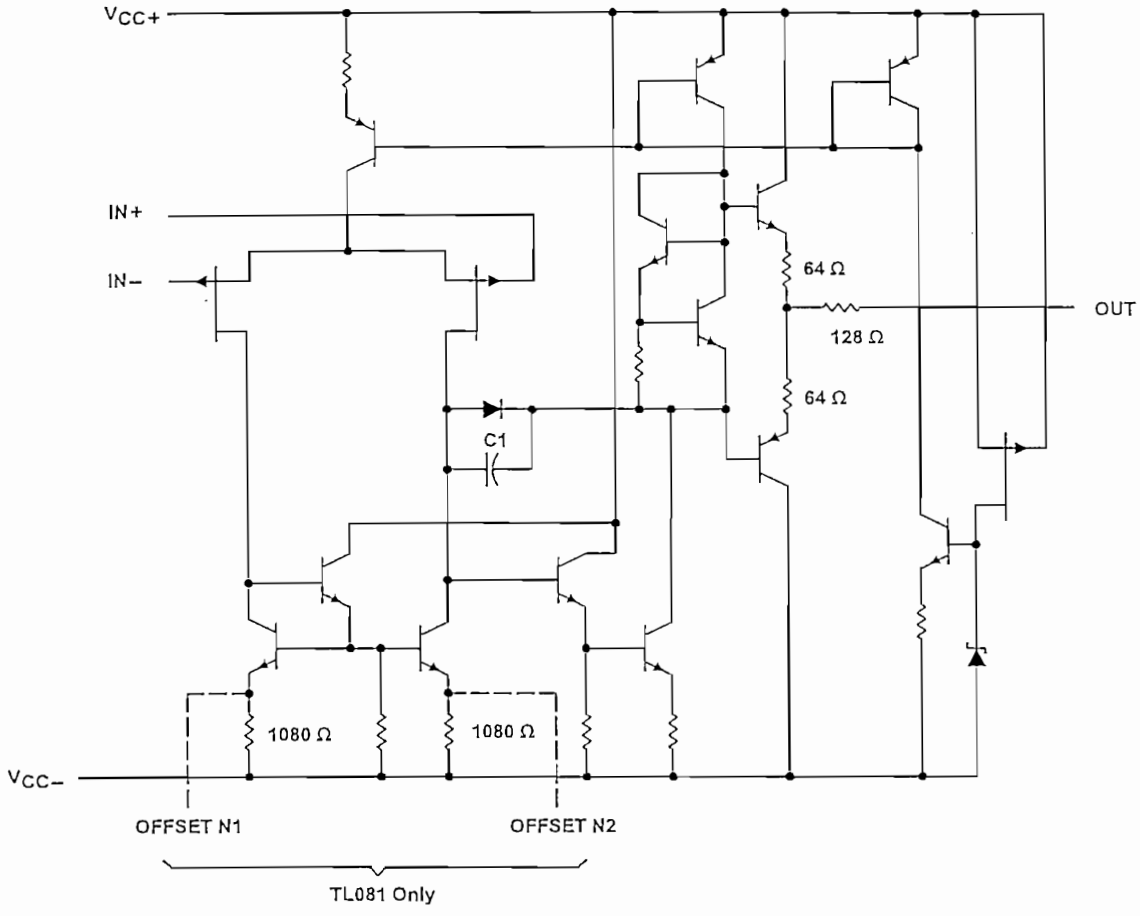
NC – No internal connection

AVAILABLE OPTIONS

T_A	V_{IO}max AT 25°C	PACKAGED DEVICES										CHIP FORM (Y)
		SMALL OUTLINE (D008)	SMALL OUTLINE (D014)	CHIP CARRIER (FK)	CERAMIC DIP (J)	CERAMIC DIP (JG)	PLASTIC DIP (N)	PLASTIC DIP (P)	TSSOP (PW)	FLAT PACK (U)	FLAT PACK (W)	
0°C to 70°C	15 mV	TL081CD	—	—	—	—	—	TL081CP	TL081CPW	—	—	—
	6 mV	TL081ACD						TL081ACP				
	3 mV	TL081BCD						TL081BCP				
	15 mV	TL082CD	—	—	—	—	—	TL082CP	TL082CPW	—	—	TL082Y
	6 mV	TL082ACD						TL082ACP				
	3 mV	TL082BCD						TL082BCP				
	15 mV	—	TL084CD	—	—	—	TL084CN	—	TL084CPW	—	—	TL084Y
	6 mV		TL084ACD				TL084ACN					
	3 mV		TL084BCD				TL084BCN					
−40°C to 85°C	6 mV	TL081ID	—	—	—	—	—	TL081IP	—	—	—	—
	6 mV	TL082ID						TL082IP				
	6 mV	TL084ID	TL084ID				TL084IN					
−55°C to 125°C	6 mV	—	—	TL081MFK	—	TL081MJG	—	—	—	TL081MU	—	—
	6 mV			TL082MFK		TL082MJG				TL082MU		
	9 mV			TL084MFK	TL084MJ						TL084MW	

The D package is available taped and reeled. Add *R* suffix to the device type (e.g., TL081CDR).

schematic (each amplifier)

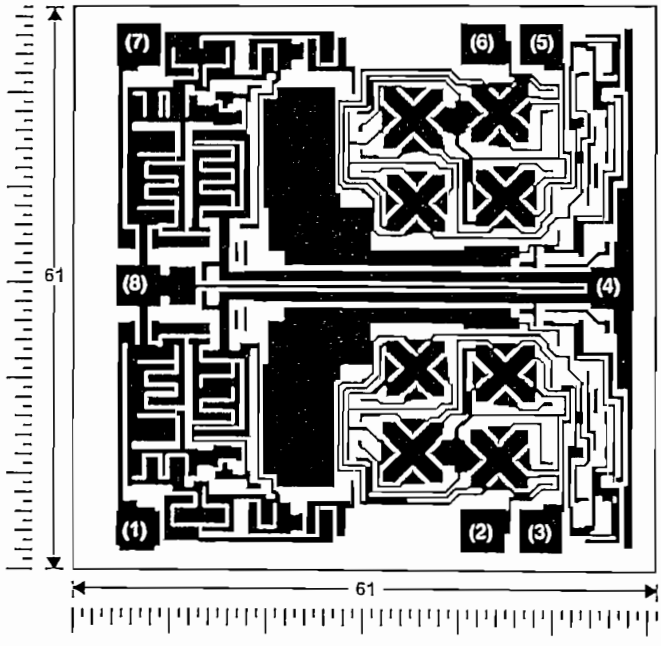


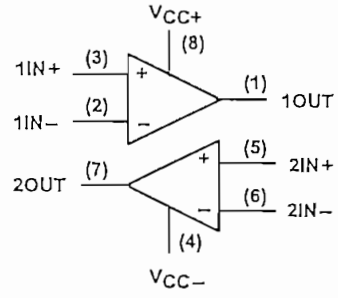
Component values shown are nominal.

TL082Y chip information

These chips, when properly assembled, display characteristics similar to the TL082. Thermal compression or ultrasonic bonding may be used on the doped-aluminum bonding pads. Chips may be mounted with conductive epoxy or a gold-silicon preform.

BONDING PAD ASSIGNMENTS





CHIP THICKNESS: 15 TYPICAL

BONDING PADS: 4 × 4 MINIMUM

$T_J$max = 150°C

TOLERANCES ARE ±10%.

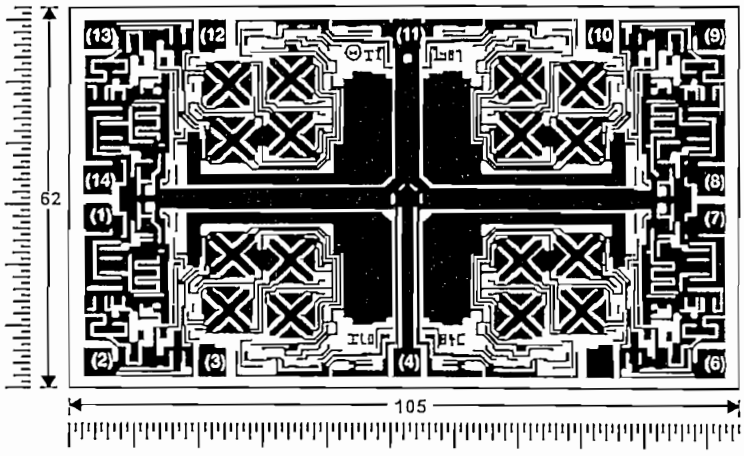
ALL DIMENSIONS ARE IN MILS.

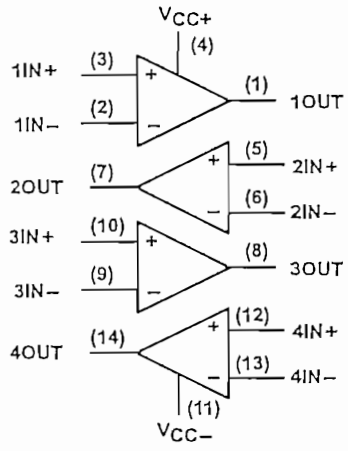
PIN (4) IS INTERNALLY CONNECTED TO BACKSIDE OF CHIP.

TL084Y chip information

These chips, when properly assembled, display characteristics similar to the TL084. Thermal compression or ultrasonic bonding may be used on the doped-aluminum bonding pads. Chips may be mounted with conductive epoxy or a gold-silicon preform.

BONDING PAD ASSIGNMENTS





CHIP THICKNESS: 15 TYPICAL

BONDING PADS: 4 x 4 MINIMUM

$T_J$max = 150°C

TOLERANCES ARE ±10%.

ALL DIMENSIONS ARE IN MILS.

PIN (11) IS INTERNALLY CONNECTED TO BACKSIDE OF CHIP.

absolute maximum ratings over operating free-air temperature range (unless otherwise noted)†

		TL08_C TL08_AC TL08_BC	TL08_I	TL08_M	UNIT
Supply voltage, V_{CC+} (see Note 1)		18	18	18	V
Supply voltage V_{CC-} (see Note 1)		−18	−18	−18	V
Differential input voltage, V_{ID} (see Note 2)		±30	±30	±30	V
Input voltage, V_I (see Notes 1 and 3)		±15	±15	±15	V
Duration of output short circuit (see Note 4)		unlimited	unlimited	unlimited	
Continuous total power dissipation		See Dissipation Rating Table			
Operating free-air temperature range, T_A		0 to 70	−40 to 85	−55 to 125	°C
Storage temperature range, T_{stg}		−65 to 150	−65 to 150	−65 to 150	°C
Case temperature for 60 seconds, T_C	FK package			260	°C
Lead temperature 1,6 mm (1/16 inch) from case for 60 seconds	J or JG package			300	°C
Lead temperature 1,6 mm (1/16 inch) from case for 10 seconds	D, N, P, or PW package	260	260		°C

† Stresses beyond those listed under "absolute maximum ratings" may cause permanent damage to the device. These are stress ratings only, and functional operation of the device at these or any other conditions beyond those indicated under "recommended operating conditions" is not implied. Exposure to absolute-maximum-rated conditions for extended periods may affect device reliability.

NOTES: 1. All voltage values, except differential voltages, are with respect to the midpoint between V_{CC+} and V_{CC-}.
2. Differential voltages are at IN+ with respect to IN−.
3. The magnitude of the input voltage must never exceed the magnitude of the supply voltage or 15 V, whichever is less.
4. The output may be shorted to ground or to either supply. Temperature and/or supply voltages must be limited to ensure that the dissipation rating is not exceeded.

DISSIPATION RATING TABLE

PACKAGE	$T_A \le 25°C$ POWER RATING	DERATING FACTOR	DERATE ABOVE T_A	$T_A = 70°C$ POWER RATING	$T_A = 85°C$ POWER RATING	$T_A = 125°C$ POWER RATING
D (8 pin)	680 mW	5.8 mW/°C	32°C	460 mW	373 mW	N/A
D (14 pin)	680 mW	7.6 mW/°C	60°C	604 mW	490 mW	N/A
FK	680 mW	11.0 mW/°C	88°C	680 mW	680 mW	273 mW
J	680 mW	11.0 mW/°C	88°C	680 mW	680 mW	273 mW
JG	680 mW	8.4 mW/°C	69°C	672 mW	546 mW	210 mW
N	680 mW	9.2 mW/°C	76°C	680 mW	597 mW	N/A
P	680 mW	8.0 mW/°C	65°C	640 mW	520 mW	N/A
PW (8 pin)	525 mW	4.2 mW/°C	25°C	336 mW	N/A	N/A
PW (14 pin)	700 mW	5.6 mW/°C	25°C	448 mW	N/A	N/A
U	675 mW	5.4 mW/°C	25°C	432 mW	351 mW	135 mW
W	680 mW	8.0 mW/°C	65°C	640 mW	520 mW	200 mW

electrical characteristics, $V_{CC\pm}$ = ±15 V (unless otherwise noted)

PARAMETER		TEST CONDITIONS		T_A†	TL081C, TL082C, TL084C			TL081AC, TL082AC, TL084AC			TL081BC, TL082BC, TL084BC			TL081I, TL082I, TL084I			UNIT
					MIN	TYP	MAX	MIN	TYP	MAX	MIN	TYP	MAX	MIN	TYP	MAX	
V_{IO}	Input offset voltage	V_O = 0	R_S = 50 Ω	25°C		3	15		3	6		2	3		3	6	mV
				Full range			20			7.5			5			9	
α_{VIO}	Temperature coefficient of input offset voltage	V_O = 0	R_S = 50 Ω	Full range		18			18			18			18		µV/°C
I_{IO}	Input offset current‡	V_O = 0		25°C		5	200		5	100		5	100		5	100	pA
				Full range			2			2			2			10	nA
I_{IB}	Input bias current‡	V_O = 0		25°C		30	400		30	200		30	200		30	200	pA
				Full range			10			7			7			20	nA
V_{ICR}	Common-mode input voltage range			25°C	±11	−12 to 15		±11	−12 to 15		±11	−12 to 15		±11	−12 to 15		V
V_{OM}	Maximum peak output voltage swing	R_L = 10 kΩ		25°C	±12	±13.5		±12	±13.5		±12	±13.5		±12	±13.5		V
		R_L ≥ 10 kΩ		Full range	±12			±12			±12			±12			
		R_L ≥ 2 kΩ			±10	±12		±10	±12		±10	±12		±10	±12		
A_{VD}	Large-signal differential voltage amplification	V_O = ±10 V,	R_L ≥ 2 kΩ	25°C	25	200		50	200		50	200		50	200		V/mV
		V_O = ±10 V,	R_L ≥ 2 kΩ	Full range	15			25			25			25			
B_1	Unity-gain bandwidth			25°C		3			3			3			3		MHz
r_i	Input resistance			25°C		10^{12}			10^{12}			10^{12}			10^{12}		Ω
CMRR	Common-mode rejection ratio	V_{IC} = V_{ICR}min, V_O = 0,	R_S = 50 Ω	25°C	70	86		75	86		75	86		75	86		dB
k_{SVR}	Supply voltage rejection ratio ($\Delta V_{CC\pm}/\Delta V_{IO}$)	V_{CC} = ±15 V to ± 9 V, V_O = 0,	R_S = 50 Ω	25°C	70	86		80	86		80	86		80	86		dB
I_{CC}	Supply current (per amplifier)	V_O = 0,	No load	25°C		1.4	2.8		1.4	2.8		1.4	2.8		1.4	2.8	mA
V_{O1}/V_{O2}	Crosstalk attenuation	A_{VD} = 100		25°C		120			120			120			120		dB

† All characteristics are measured under open-loop conditions with zero common-mode voltage unless otherwise specified. Full range for T_A is 0°C to 70°C for TL08_C, TL08_AC, TL08_BC and −40°C to 85°C for TL08_I.

‡ Input bias currents of a FET-input operational amplifier are normal junction reverse currents, which are temperature sensitive as shown in Figure 17. Pulse techniques must be used that maintain the junction temperature as close to the ambient temperature as possible.

electrical characteristics, $V_{CC\pm} = \pm 15$ V (unless otherwise noted)

PARAMETER		TEST CONDITIONS†		T_A	TL081M, TL082M			TL084M			UNIT
					MIN	TYP	MAX	MIN	TYP	MAX	
V_{IO}	Input offset voltage	$V_O = 0$,	$R_S = 50\ \Omega$	25°C		3	6		3	9	mV
				–55°C to 125°C			9			15	
α_{VIO}	Temperature coefficient of input offset voltage	$V_O = 0$	$R_S = 50\ \Omega$	–55°C to 125°C		18			18		µV/°C
I_{IO}	Input offset current‡	$V_O = 0$		25°C		5	100		5	100	pA
				125°C			20			20	nA
I_{IB}	Input bias current‡	$V_O = 0$		25°C		30	200		30	200	pA
				125°C			50			50	nA
V_{ICR}	Common-mode input voltage range			25°C	±11	±12 to 15		±11	±12 to 15		V
V_{OM}	Maximum peak output voltage swing	$R_L = 10$ kΩ		25°C	±12	±13.5		±12	±13.5		V
		$R_L \geq 10$ kΩ		–55°C to 125°C	±12			±12			
		$R_L \geq 2$ kΩ			±10	±12		±10	±12		
A_{VD}	Large-signal differential voltage amplification	$V_O = \pm 10$ V,	$R_L \geq 2$ kΩ	25°C	25	200		25	200		V/mV
		$V_O = \pm 10$ V,	$R_L \geq 2$ kΩ	–55°C to 125°C	15			15			
B_1	Unity-gain bandwidth			25°C		3			3		MHz
r_i	Input resistance			25°C		10^{12}			10^{12}		Ω
CMRR	Common-mode rejection ratio	$V_{IC} = V_{ICR}$min, $V_O = 0$,	$R_S = 50\ \Omega$	25°C	80	86		80	86		dB
k_{SVR}	Supply voltage rejection ratio ($\Delta V_{CC\pm}/\Delta V_{IO}$)	$V_{CC} = \pm 15$ V to ±9 V, $V_O = 0$,	$R_S = 50\ \Omega$	25°C	80	86		80	86		dB
I_{CC}	Supply current (per amplifier)	$V_O = 0$,	No load	25°C		1.4	2.8		1.4	2.8	mA
V_{O1}/V_{O2}	Crosstalk attenuation	$A_{VD} = 100$		25°C		120			120		dB

† All characteristics are measured under open-loop conditions with zero common-mode input voltage unless otherwise specified.
‡ Input bias currents of a FET-input operational amplifier are normal junction reverse currents, which are temperature sensitive as shown in Figure 17. Pulse techniques must be used that maintain the junction temperatures as close to the ambient temperature as is possible.

operating characteristics, $V_{CC\pm} = \pm 15$ V, $T_A = 25$°C (unless otherwise noted)

PARAMETER		TEST CONDITIONS			MIN	TYP	MAX	UNIT
SR	Slew rate at unity gain	$V_I = 10$ V,	$R_L = 2$ kΩ, $C_L = 100$ pF,	See Figure 1	8*	13		V/µs
		$V_I = 10$ V, $T_A = -55$°C to 125°C,	$R_L = 2$ kΩ, $C_L = 100$ pF,	See Figure 1	5*			
t_r	Rise time	$V_I = 20$ mV,	$R_L = 2$ kΩ, $C_L = 100$ pF,	See Figure 1		0.05		µs
	Overshoot factor					20%		
V_n	Equivalent input noise voltage	$R_S = 20\ \Omega$	f = 1 kHz			18		nV/√Hz
			f = 10 Hz to 10 kHz			4		µV
I_n	Equivalent input noise current	$R_S = 20\ \Omega$,	f = 1 kHz			0.01		pA/√Hz
THD	Total harmonic distortion	$V_{O(rms)} = 10$ V,	$R_S \leq 1$ kΩ, $R_L \geq 2$ kΩ,	f = 1 kHz		0.003%		

*On products compliant to MIL-PRF-38535, this parameter is not production tested.

electrical characteristics, $V_{CC\pm} = \pm 15$ V, $T_A = 25°C$ (unless otherwise noted)

PARAMETER		TEST CONDITIONS†		TL082Y, TL084Y			UNIT
				MIN	TYP	MAX	
V_{IO}	Input offset voltage	$V_O = 0$,	$R_S = 50\ \Omega$		3	15	mV
α_{VIO}	Temperature coefficient of input offset voltage	$V_O = 0$,	$R_S = 50\ \Omega$		18		µV/°C
I_{IO}	Input offset current‡	$V_O = 0$,			5	200	pA
I_{IB}	Input bias current‡	$V_O = 0$,			30	400	pA
V_{ICR}	Common-mode input voltage range			±11	−12 to 15		V
V_{OM}	Maximum peak output voltage swing	$R_L = 10\ k\Omega$,		±12	±13.5		V
A_{VD}	Large-signal differential voltage amplification	$V_O = \pm 10$ V,	$R_L \geq 2\ k\Omega$	25	200		V/mV
B_1	Unity-gain bandwidth				3		MHz
r_i	Input resistance				10^{12}		Ω
CMRR	Common-mode rejection ratio	$V_{IC} = V_{ICR}$min, $R_S = 50\ \Omega$	$V_O = 0$,	70 70	86 86		dB
k_{SVR}	Supply voltage rejection ratio ($\Delta V_{CC\pm}/\Delta V_{IO}$)	$V_{CC} = \pm 15$ V to ± 9 V, $V_O = 0$,	$R_S = 50\ \Omega$	70 70	86 86		dB
I_{CC}	Supply current (per amplifier)	$V_O = 0$,	No load		1.4	2.8	mA
V_{O1}/V_{O2}	Crosstalk attenuation	$A_{VD} = 100$			120		dB

† All characteristics are measured under open-loop conditions with zero common-mode voltage unless otherwise specified.
‡ Input bias currents of a FET-input operational amplifier are normal junction reverse currents, which are temperature sensitive as shown in Figure 17. Pulse techniques must be used that maintain the junction temperature as close to the ambient temperature as possible.

operating characteristics, $V_{CC\pm} = \pm 15$ V, $T_A = 25°C$

PARAMETER		TEST CONDITIONS			MIN	TYP	MAX	UNIT
SR	Slew rate at unity gain	$V_I = 10$ V,	$R_L = 2\ k\Omega$, $C_L = 100$ pF,	See Figure 1	8	13		V/µs
t_r	Rise time	$V_I = 20$ mV,	$R_L = 2\ k\Omega$, $C_L = 100$ pF,	See Figure 1		0.05		µs
	Overshoot factor					20%		
V_n	Equivalent input noise voltage	$R_S = 20\ \Omega$	f = 1 kHz			18		nV/√Hz
			f = 10 Hz to 10 kHz			4		µV
I_n	Equivalent input noise current	$R_S = 20\ \Omega$,	f = 1 kHz			0.01		pA/√Hz
THD	Total harmonic distortion	$V_{O(rms)} = 10$ V,	$R_S \leq 1\ k\Omega$, $R_L \geq 2\ k\Omega$,	f = 1 kHz		0.003%		

PARAMETER MEASUREMENT INFORMATION

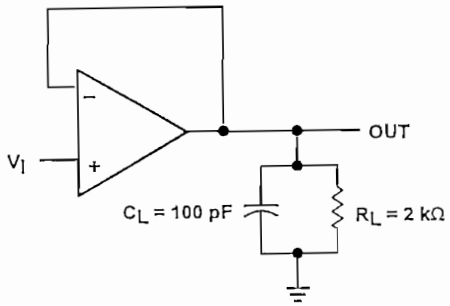


Figure 1

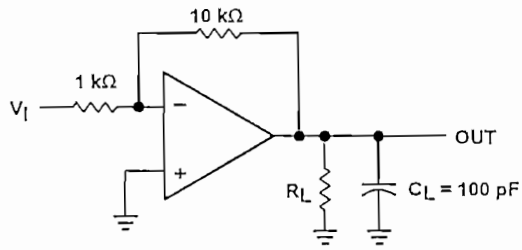


Figure 2

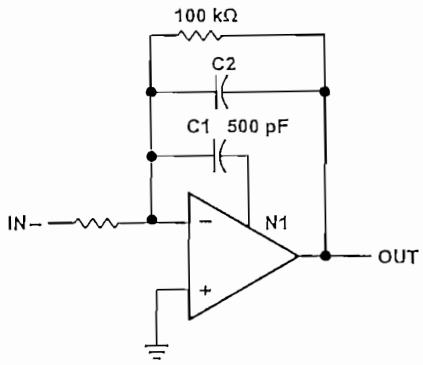


Figure 3

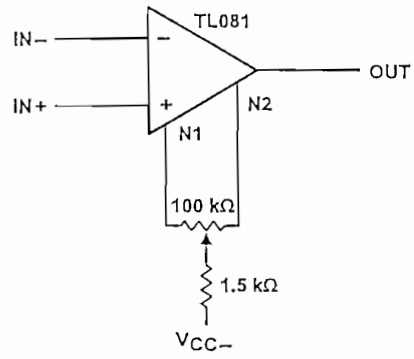


Figure 4

TYPICAL CHARACTERISTICS

Table of Graphs

			FIGURE
V_{OM}	Maximum peak output voltage	vs Frequency vs Free-air temperature vs Load resistance vs Supply voltage	5, 6, 7 8 9 10
A_{VD}	Large-signal differential voltage amplification	vs Free-air temperature vs Frequency	11 12
	Differential voltage amplification	vs Frequency with feed-forward compensation	13
P_D	Total power dissipation	vs Free-air temperature	14
I_{CC}	Supply current	vs Free-air temperature vs Supply voltage	15 16
I_{IB}	Input bias current	vs Free-air temperature	17
	Large-signal pulse response	vs Time	18
V_O	Output voltage	vs Elapsed time	19
CMRR	Common-mode rejection ratio	vs Free-air temperature	20
V_n	Equivalent input noise voltage	vs Frequency	21
THD	Total harmonic distortion	vs Frequency	22

MAXIMUM PEAK OUTPUT VOLTAGE
vs
FREQUENCY

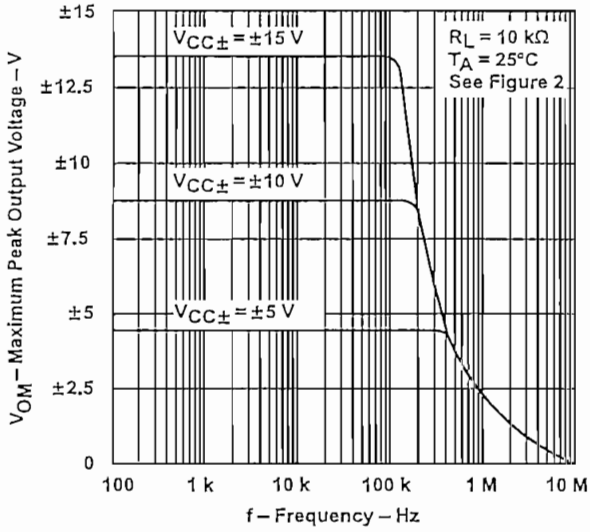


Figure 5

MAXIMUM PEAK OUTPUT VOLTAGE
vs
FREQUENCY

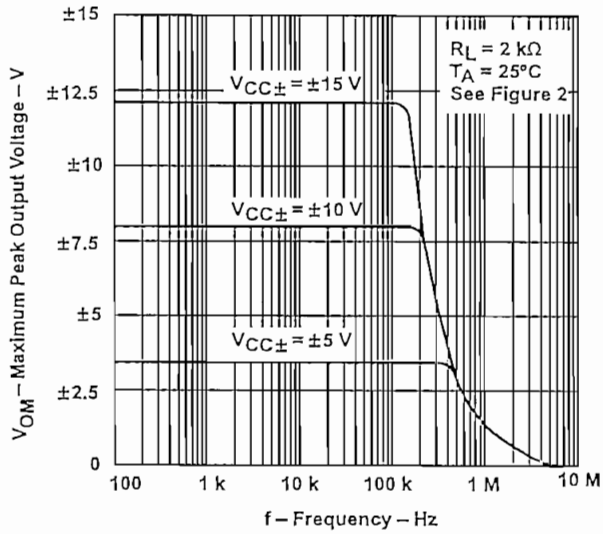


Figure 6

TYPICAL CHARACTERISTICS†

MAXIMUM PEAK OUTPUT VOLTAGE
vs
FREQUENCY

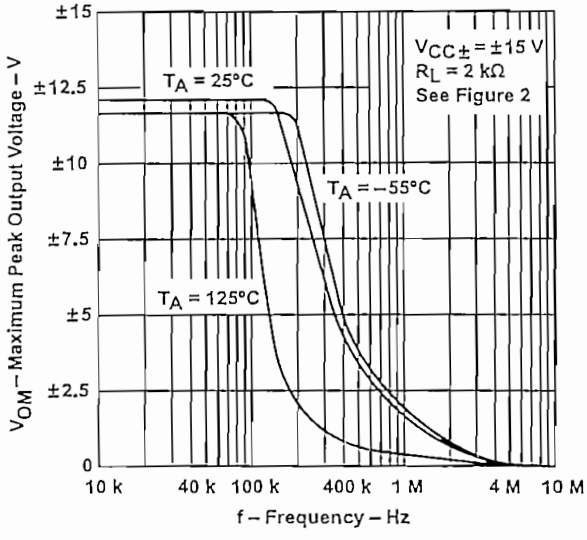


Figure 7

MAXIMUM PEAK OUTPUT VOLTAGE
vs
FREE-AIR TEMPERATURE

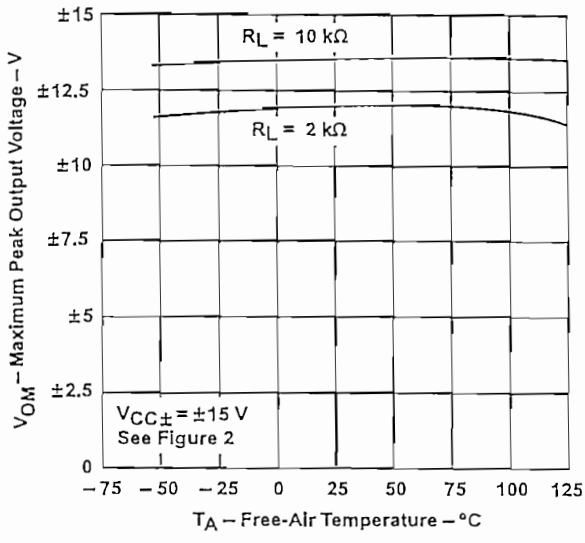


Figure 8

MAXIMUM PEAK OUTPUT VOLTAGE
vs
LOAD RESISTANCE

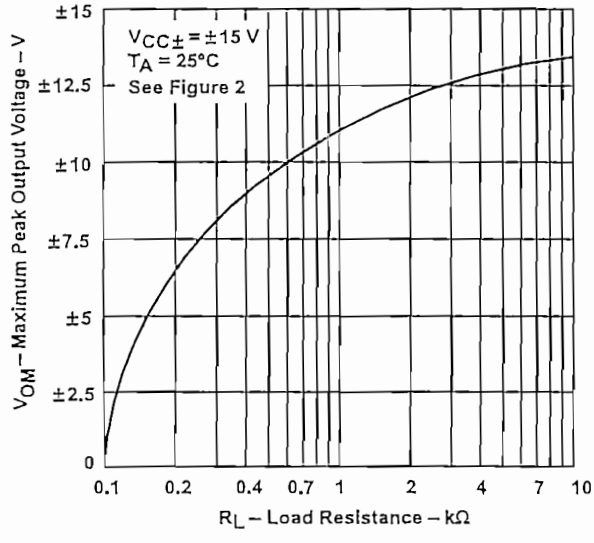


Figure 9

MAXIMUM PEAK OUTPUT VOLTAGE
vs
SUPPLY VOLTAGE

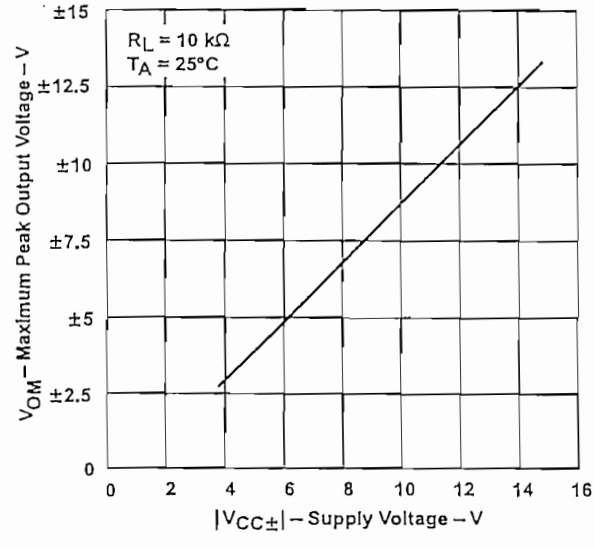


Figure 10

† Data at high and low temperatures are applicable only within the rated operating free-air temperature ranges of the various devices.

TYPICAL CHARACTERISTICS†

LARGE-SIGNAL
DIFFERENTIAL VOLTAGE AMPLIFICATION
vs
FREE-AIR TEMPERATURE

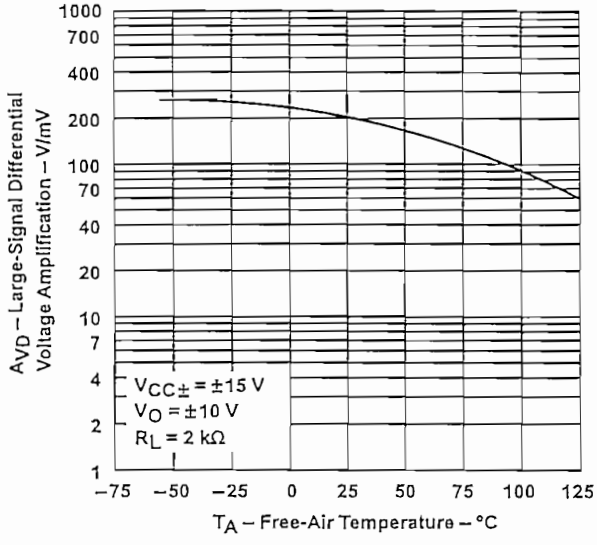


Figure 11

LARGE-SIGNAL
DIFFERENTIAL VOLTAGE AMPLIFICATION
vs
FREQUENCY

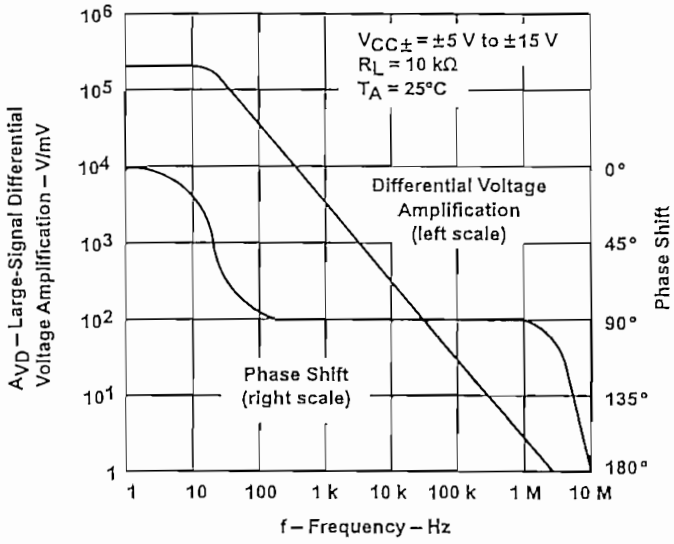


Figure 12

† Data at high and low temperatures are applicable only within the rated operating free-air temperature ranges of the various devices.

TYPICAL CHARACTERISTICS†

DIFFERENTIAL VOLTAGE AMPLIFICATION
vs
FREQUENCY WITH FEED-FORWARD COMPENSATION

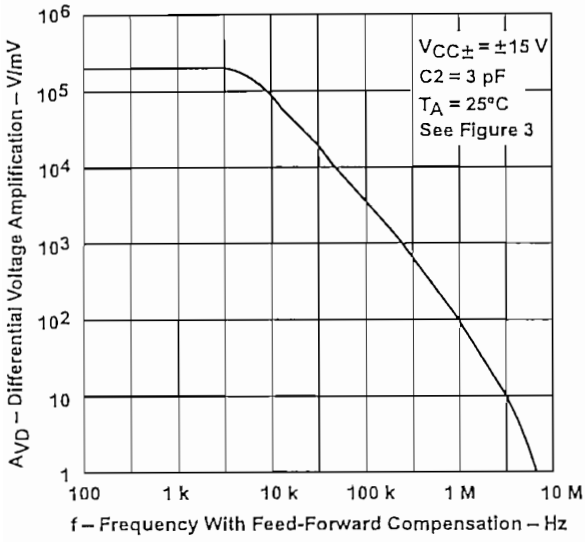


Figure 13

TOTAL POWER DISSIPATION
vs
FREE-AIR TEMPERATURE

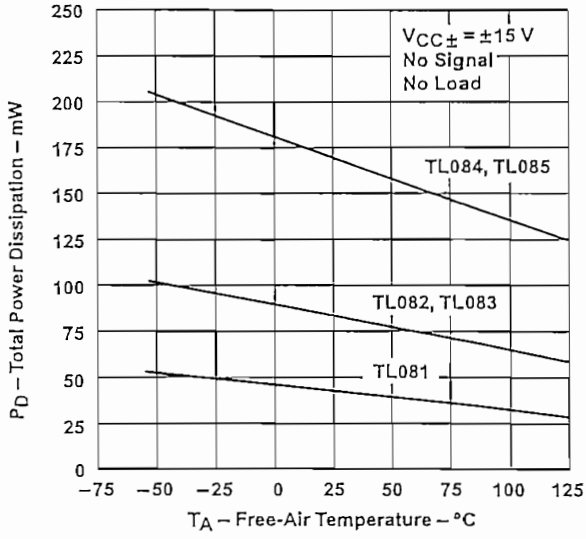


Figure 14

SUPPLY CURRENT PER AMPLIFIER
vs
FREE-AIR TEMPERATURE

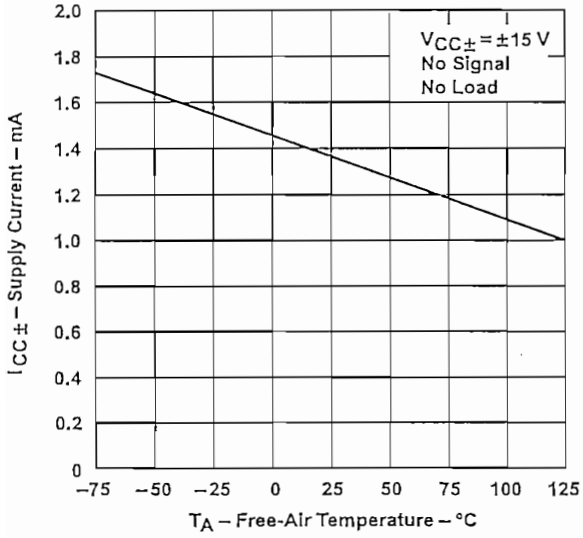


Figure 15

SUPPLY CURRENT
vs
SUPPLY VOLTAGE

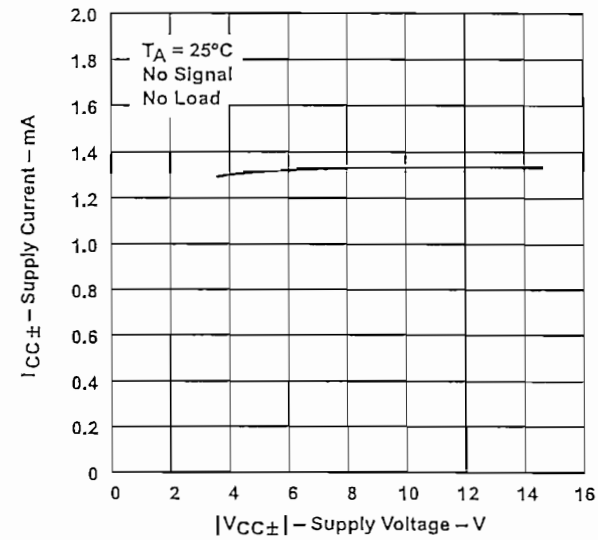


Figure 16

† Data at high and low temperatures are applicable only within the rated operating free-air temperature ranges of the various devices.

TYPICAL CHARACTERISTICS†

INPUT BIAS CURRENT
vs
FREE-AIR TEMPERATURE

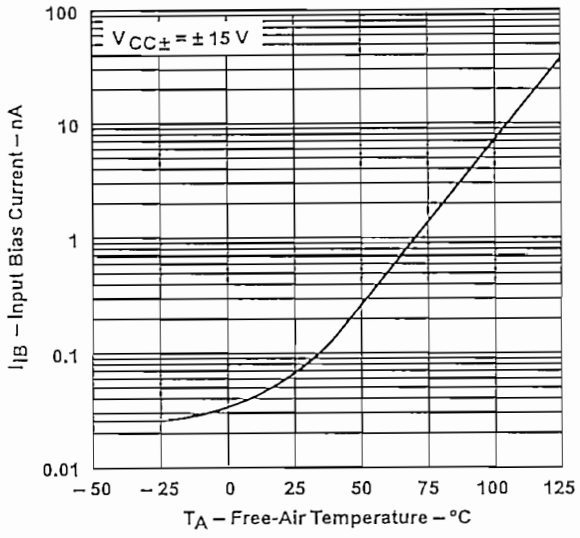


Figure 17

VOLTAGE-FOLLOWER
LARGE-SIGNAL PULSE RESPONSE

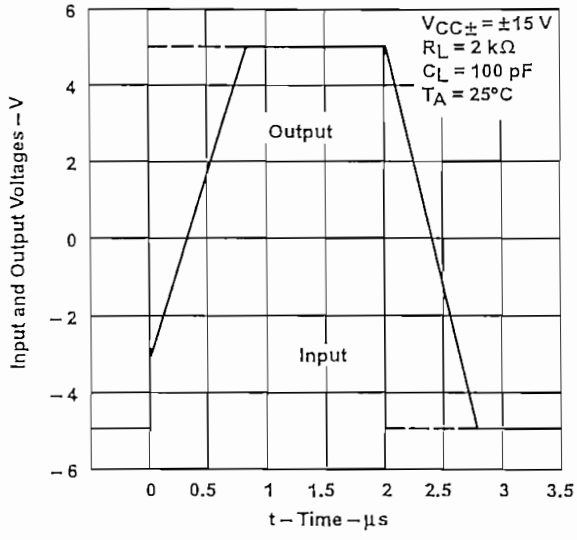


Figure 18

OUTPUT VOLTAGE
vs
ELAPSED TIME

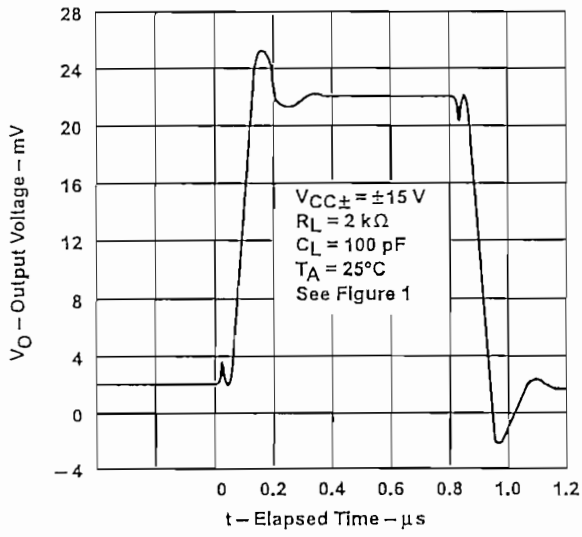


Figure 19

COMMON-MODE REJECTION RATIO
vs
FREE-AIR TEMPERATURE

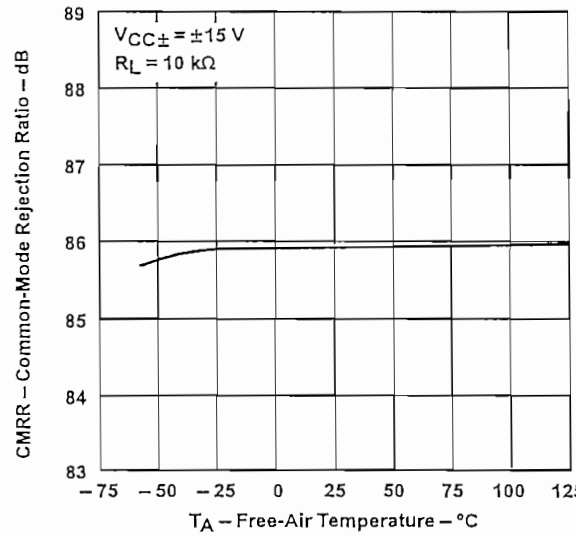


Figure 20

† Data at high and low temperatures are applicable only within the rated operating free-air temperature ranges of the various devices.

TYPICAL CHARACTERISTICS†

EQUIVALENT INPUT NOISE VOLTAGE
vs
FREQUENCY

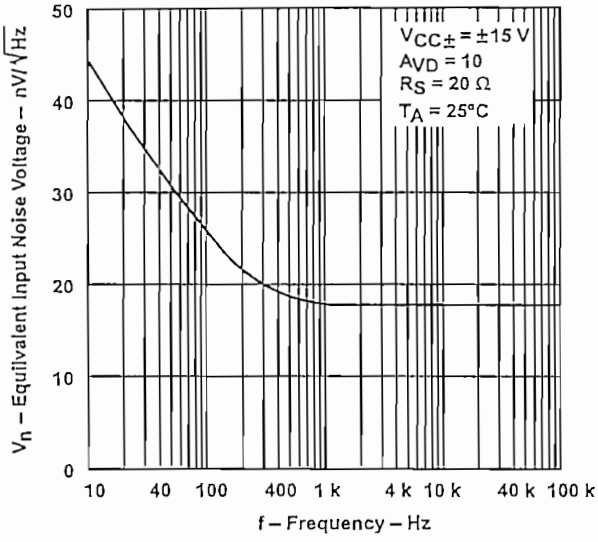


Figure 21

TOTAL HARMONIC DISTORTION
vs
FREQUENCY

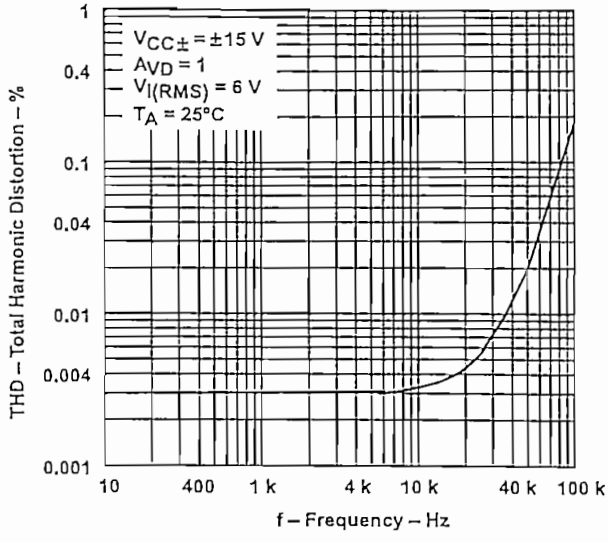


Figure 22

† Data at high and low temperatures are applicable only within the rated operating free-air temperature ranges of the various devices.

APPLICATION INFORMATION

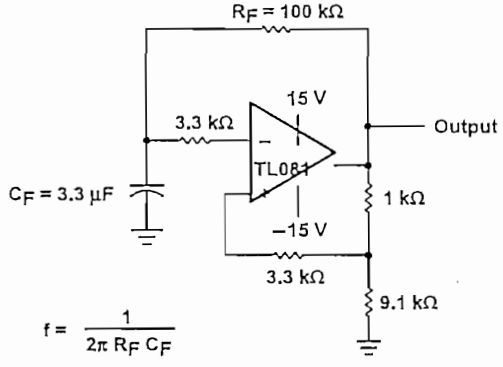


Figure 23

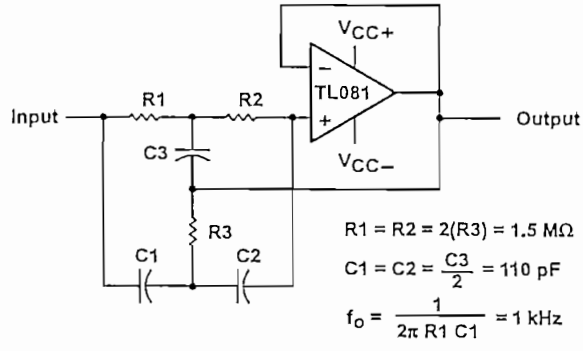


Figure 24

APPLICATION INFORMATION

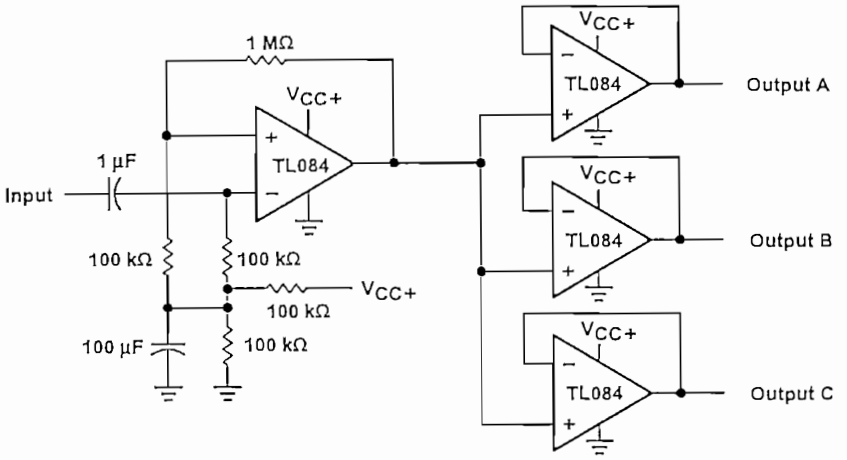
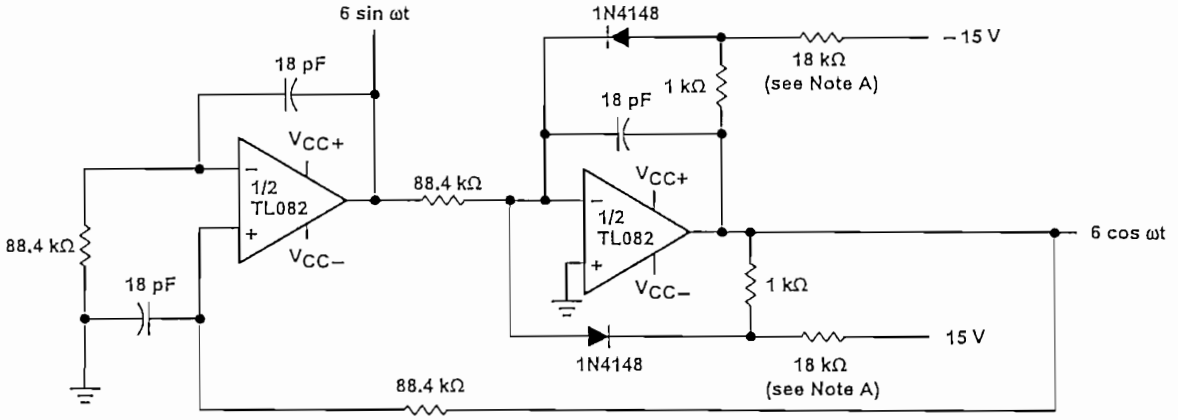


Figure 25. Audio-Distribution Amplifier



NOTE A: These resistor values may be adjusted for a symmetrical output.

Figure 26. 100-KHz Quadrature Oscillator

APPLICATION INFORMATION

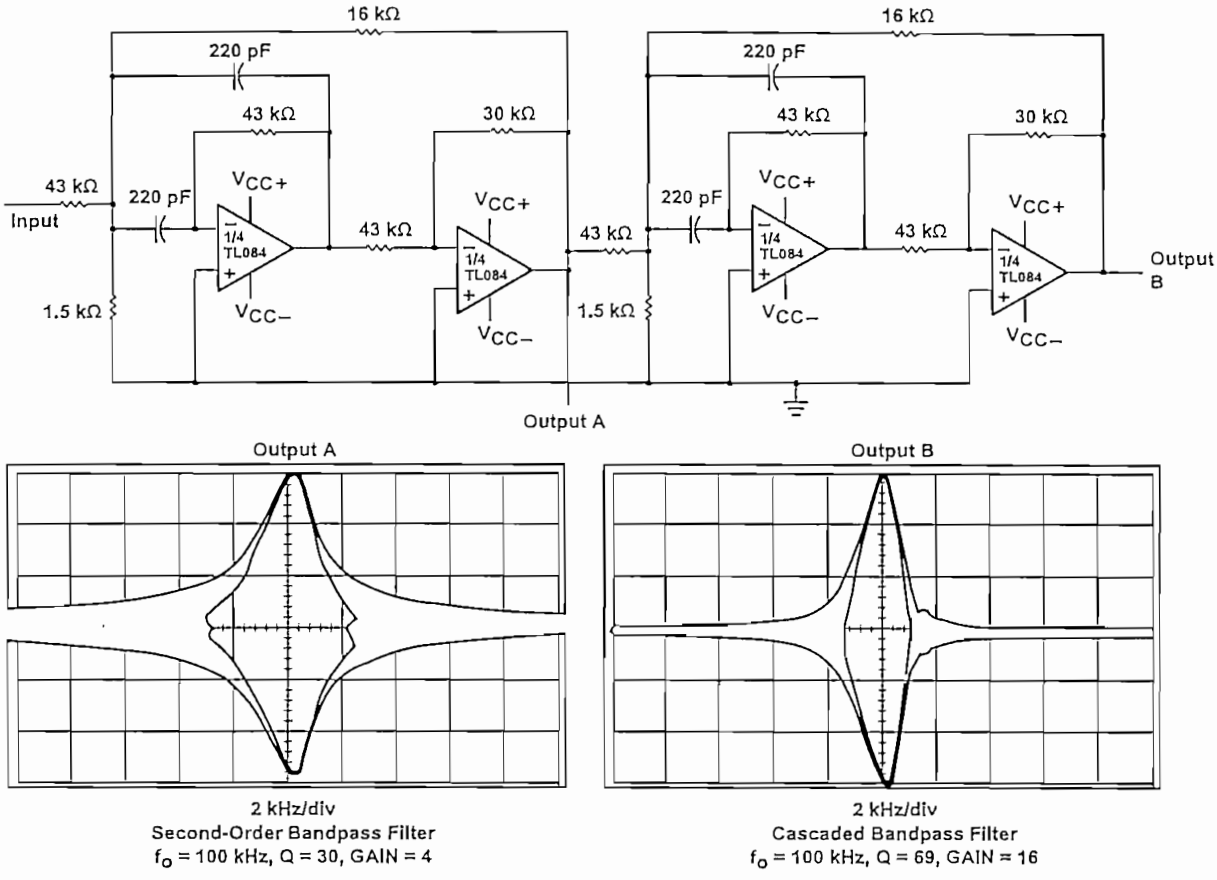


Figure 27. Positive-Feedback Bandpass Filter

IMPORTANT NOTICE

Texas Instruments (TI) reserves the right to make changes to its products or to discontinue any semiconductor product or service without notice, and advises its customers to obtain the latest version of relevant information to verify, before placing orders, that the information being relied on is current.

TI warrants performance of its semiconductor products and related software to the specifications applicable at the time of sale in accordance with TI's standard warranty. Testing and other quality control techniques are utilized to the extent TI deems necessary to support this warranty. Specific testing of all parameters of each device is not necessarily performed, except those mandated by government requirements.

Certain applications using semiconductor products may involve potential risks of death, personal injury, or severe property or environmental damage ("Critical Applications").

TI SEMICONDUCTOR PRODUCTS ARE NOT DESIGNED, INTENDED, AUTHORIZED, OR WARRANTED TO BE SUITABLE FOR USE IN LIFE-SUPPORT APPLICATIONS, DEVICES OR SYSTEMS OR OTHER CRITICAL APPLICATIONS.

Inclusion of TI products in such applications is understood to be fully at the risk of the customer. Use of TI products in such applications requires the written approval of an appropriate TI officer. Questions concerning potential risk applications should be directed to TI through a local SC sales office.

In order to minimize risks associated with the customer's applications, adequate design and operating safeguards should be provided by the customer to minimize inherent or procedural hazards.

TI assumes no liability for applications assistance, customer product design, software performance, or infringement of patents or services described herein. Nor does TI warrant or represent that any license, either express or implied, is granted under any patent right, copyright, mask work right, or other intellectual property right of TI covering or relating to any combination, machine, or process in which such semiconductor products or services might be or are used.

Copyright © 1996, Texas Instruments Incorporated

XICOR®

Digitally Controlled Potentiometer (XDCP™) X9C102/103/104/503

FEATURES

- Solid-state potentiometer
- 3-wire serial interface
- 100 wiper tap points
  - —Wiper position stored in nonvolatile memory and recalled on power-up
- 99 resistive elements
  - —Temperature compensated
  - —End to end resistance, ±20%
  - —Terminal voltages, ±5V
- Low power CMOS
  - —V_{CC} = 5V
  - —Active current, 3mA max.
  - —Standby current, 750µA max.
- High reliability
  - —Endurance, 100,000 data changes per bit
  - —Register data retention, 100 years
- X9C102 = 1 kΩ
- X9C103 = 10 kΩ
- X9C503 = 50 kΩ
- X9C104 = 100 kΩ
- Packages
  - —8-lead SOIC and DIP

DESCRIPTION

The X9Cxxx are Xicor digitally controlled (XDCP) potentiometers. The device consists of a resistor array, wiper switches, a control section, and nonvolatile memory. The wiper position is controlled by a three-wire interface.

The potentiometer is implemented by a resistor array composed of 99 resistive elements and a wiper switching network. Between each element and at either end are tap points accessible to the wiper terminal. The position of the wiper element is controlled by the $\overline{CS}$, U/$\overline{D}$, and $\overline{INC}$ inputs. The position of the wiper can be stored in nonvolatile memory and then be recalled upon a subsequent power-up operation.

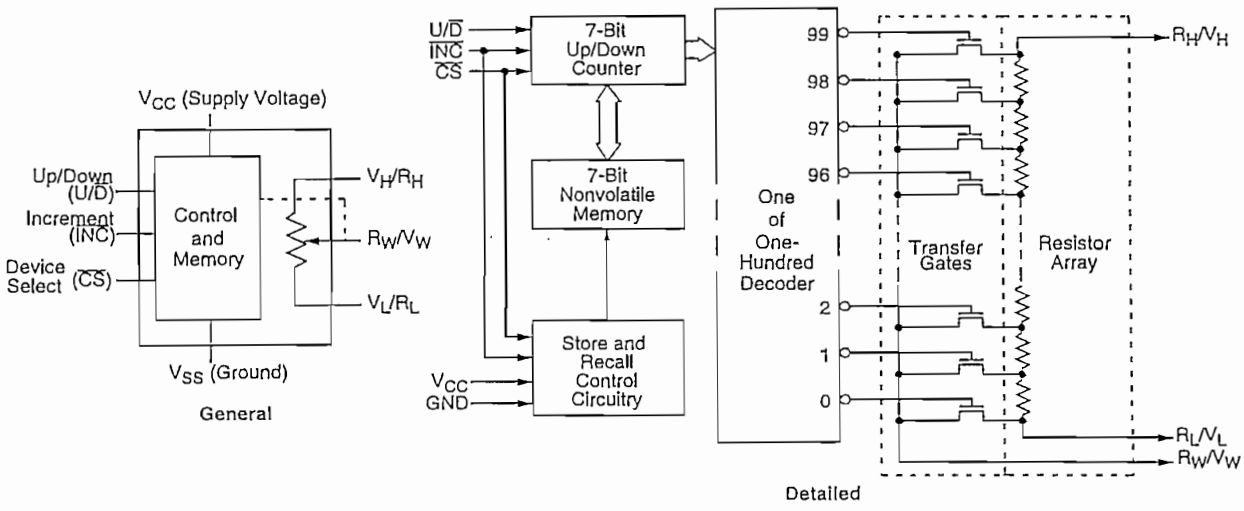
The device can be used as a three-terminal potentiometer or as a two-terminal variable resistor in a wide variety of applications including:

– control

– parameter adjustments

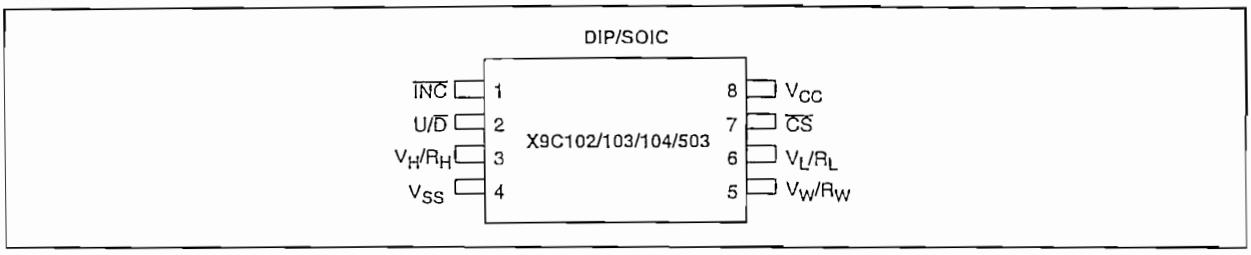
– signal processing

BLOCK DIAGRAM



XDCP is a trademark of Xicor, Inc.

PIN CONFIGURATION



ORDERING INFORMATION

Part Number	Temperature Range	Package	End to End Resistance
X9C102P	0°C TO +70°C	8-Lead Plastic DIP	1 kΩ
X9C102PI	-40°C TO +85°C	8-Lead Plastic DIP	1 kΩ
X9C102S	0°C TO +70°C	8-Lead SOIC	1 kΩ
X9C102SI	-40°C TO +85°C	8-Lead SOIC	1 kΩ
X9C103P	0°C TO +70°C	8-Lead Plastic DIP	10 kΩ
X9C103PI	-40°C TO +85°C	8-Lead Plastic DIP	10 kΩ
X9C103S	0°C TO +70°C	8-Lead SOIC	10 kΩ
X9C103SI	-40°C TO +85°C	8-Lead SOIC	10 kΩ
X9C104P	0°C TO +70°C	8-Lead Plastic DIP	100 kΩ
X9C104PI	-40°C TO +85°C	8-Lead Plastic DIP	100 kΩ
X9C104S	0°C TO +70°C	8-Lead SOIC	100 kΩ
X9C104SI	-40°C TO +85°C	8-Lead SOIC	100 kΩ
X9C503P	0°C TO +70°C	8-Lead Plastic DIP	50 kΩ
X9C503PI	-40°C TO +85°C	8-Lead Plastic DIP	50 kΩ
X9C503S	0°C TO +70°C	8-Lead SOIC	50 kΩ
X9C503SI	-40°C TO +85°C	8-Lead SOIC	50 kΩ

PIN DESCRIPTIONS

Pin	Symbol	Brief Description
1	$\overline{INC}$	Increment. The $\overline{INC}$ input is negative-edge triggered. Toggling $\overline{INC}$ will move the wiper and either increment or decrement the counter in the direction indicated by the logic level on the U/$\overline{D}$ input.
2	U/$\overline{D}$	Up/Down. The U/$\overline{D}$ input controls the direction of the wiper movement and whether the counter is incremented or decremented.
3	R_H/V_H	R_H/V_H. The high (V_H/R_H) terminals of the X9C102/103/104/503 are equivalent to the fixed terminals of a mechanical potentiometer. The minimum voltage is –5V and the maximum is +5V. The terminology of V_H/R_H and V_L/R_L references the relative position of the terminal in relation to wiper movement direction selected by the U/$\overline{D}$ input and not the voltage potential on the terminal.
4	V_{SS}	V_{SS}
5	V_W/R_W	V_W/R_W. V_W/R_W is the wiper terminal, and is equivalent to the movable terminal of a mechanical potentiometer. The position of the wiper within the array is determined by the control inputs. The wiper terminal series resistance is typically 40Ω.
6	R_L/V_L	R_L/V_L. The low (V_L/R_L) terminals of the X9C102/103/104/503 are equivalent to the fixed terminals of a mechanical potentiometer. The minimum voltage is –5V and the maximum is +5V. The terminology of V_H/R_H and V_L/R_L references the relative position of the terminal in relation to wiper movement direction selected by the U/$\overline{D}$ input and not the voltage potential on the terminal.
7	$\overline{CS}$	$\overline{CS}$. The device is selected when the $\overline{CS}$ input is LOW. The current counter value is stored in nonvolatile memory when $\overline{CS}$ is returned HIGH while the $\overline{INC}$ input is also HIGH. After the store operation is complete the X9C102/103/104/503 device will be placed in the low power standby mode until the device is selected once again.
8	V_{CC}	V_{CC}

ABSOLUTE MAXIMUM RATINGS

Temperature under bias .....–65°C to +135°C
Storage temperature .....–65°C to +150°C
Voltage on $\overline{CS}$, $\overline{INC}$, U/$\overline{D}$ and V_{CC}
with respect to V_{SS}.....–1V to +7V
Voltage on V_H/R_H and V_L/R_L
referenced to V_{SS} .....–8V to +8V
$\Delta V = |V_H/R_H - V_L/R_L|$
X9C102 ..... 4V
X9C103, X9C503, and X9C104 ..... 10V
Lead temperature (soldering, 10 seconds)..... +300°C
I_W (10 seconds) 8.8mA

COMMENT

Stresses above those listed under "Absolute Maximum Ratings" may cause permanent damage to the device. This is a stress rating only; functional operation of the device (at these or any other conditions above those listed in the operational sections of this specification) is not implied. Exposure to absolute maximum rating conditions for extended periods may affect device reliability.

RECOMMENDED OPERATING CONDITIONS

Temperature	Min.	Max.
Commercial	0°C	+70°C
Industrial	–40°C	+85°C

Supply Voltage (V_{CC})	Limits
X9C102/103/104/503	5V ±10%

POTENTIOMETER CHARACTERISTICS (Over recommended operating conditions unless otherwise stated.)

Symbol	Parameter	Limits			Unit	Test Conditions/Notes
		Min.	Typ.	Max.		
R_{TOTAL}	End to end resistance variation	–20		+20	%	
$V_{VH/RH}$	V_H terminal voltage	–5		+5	V	
$V_{VL/RL}$	V_L terminal voltage	–5		+5	V	
	Power rating			16	mW	X9C102
	Power rating			10	mW	X9C103/104/503
I_W	Wiper current	-4.4		4.4	mA	
R_W	Wiper resistance		40	100	Ω	Wiper Current = ±1mA
	Noise		–120		dBV	Ref. 1kHz
	Resolution		1		%	
	Absolute linearity(1)	–1		+1	MI(3)	$V_{W(n)(actual)} - V_{W(n)(expected)}$
	Relative linearity(2)	–0.2		+0.2	MI(3)	$V_{W(n+1)(actual)} - [V_{W(n)} + MI]$
	RTOTAL temperature coefficient		±300		ppm/°C	X9C103/503/104
	RTOTAL temperature coefficient		±600		ppm/°C	X9C102
	Ratiometric temperature coefficient			±20	ppm/°C	
$C_H/C_L/C_W$	Potentiometer capacitances		10/10/25		pF	See Circuit #3, Macro Model

Notes: (1) Absolute linearity is utilized to determine actual wiper voltage versus expected voltage = $[V_{W(n)(actual)} - V_{W(n)(expected)}] = \pm 1$ MI Maximum.
(2) Relative linearity is a measure of the error in step size between taps = $V_{W(n+1)} - [V_{W(n)} + MI] = +0.2$ MI.
(3) 1 MI = Minimum Increment = $R_{TOT}/99$
(4) Typical values are for $T_A = 25°C$ and nominal supply voltage.
(5) This parameter is not 100% tested.

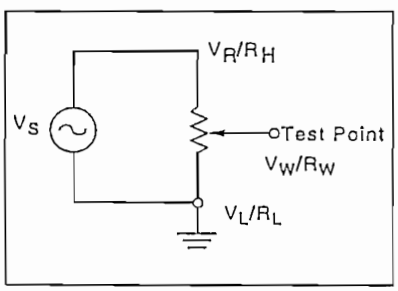
D.C. OPERATING CHARACTERISTICS (Over recommended operating conditions unless otherwise specified.)

Symbol	Parameter	Limits			Unit	Test Conditions
		Min.	Typ.(4)	Max.		
I_{CC}	V_{CC} active current		1	3	mA	$\overline{CS}$ = V_{IL}, U/$\overline{D}$ = V_{IL} or V_{IH} and $\overline{INC}$ = 0.4V to 2.4V @ max. t_{CYC}
I_{SB}	Standby supply current		200	750	µA	$\overline{CS}$ = V_{CC} – 0.3V, U/$\overline{D}$ and $\overline{INC}$ = V_{SS} or V_{CC} – 0.3V
I_{LI}	$\overline{CS}$, $\overline{INC}$, U/$\overline{D}$ input leakage current			±10	µA	V_{IN} = V_{SS} to V_{CC}
V_{IH}	$\overline{CS}$, $\overline{INC}$, U/$\overline{D}$ input HIGH voltage	2		V_{CC} + 1	V	
V_{IL}	$\overline{CS}$, $\overline{INC}$, U/$\overline{D}$ input LOW voltage	–1		0.8	V	
C_{IN}(2)	$\overline{CS}$, $\overline{INC}$, U/$\overline{D}$ input capacitance			10	pF	V_{CC} = 5V, V_{IN} = V_{SS}, T_A = 25°C, f = 1MHz

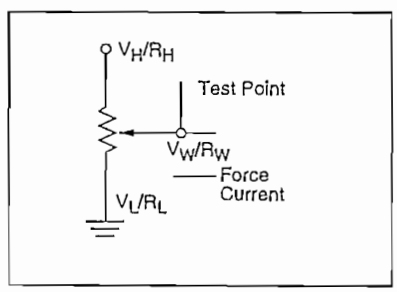
ENDURANCE AND DATA RETENTION

Parameter	Min.	Unit
Minimum endurance	100,000	Data changes per bit per register
Data retention	100	years

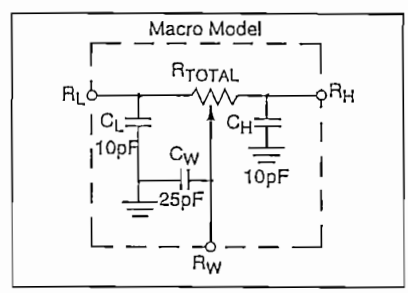
Test Circuit #1



Test Circuit #2



Test Circuit #3



A.C. CONDITIONS OF TEST

Input pulse levels	0V to 3V
Input rise and fall times	10ns
Input reference levels	1.5V

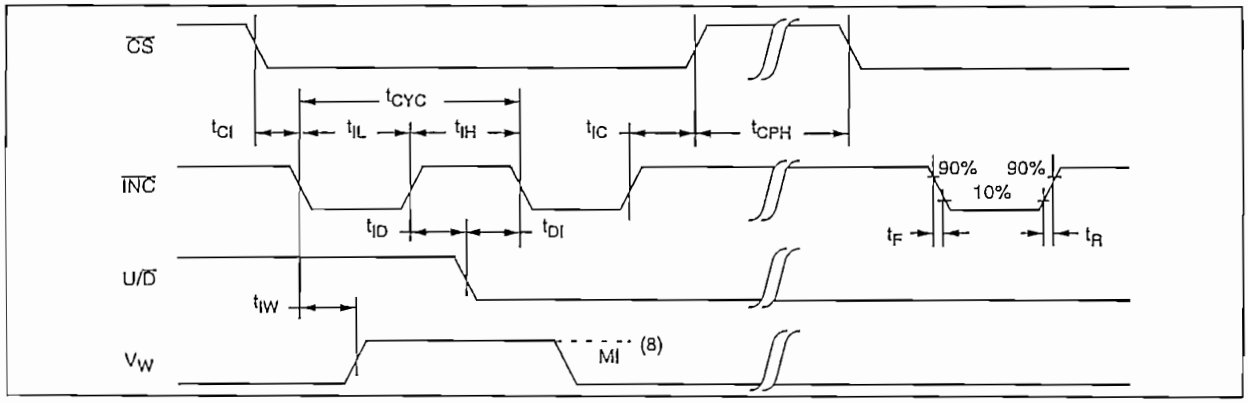
A.C. OPERATING CHARACTERISTICS (Over recommended operating conditions unless otherwise specified)

Symbol	Parameter	Limits			Unit
		Min.	Typ.(6)	Max.	
t_{CI}	$\overline{CS}$ to $\overline{INC}$ setup	100			ns
t_{ID}	$\overline{INC}$ HIGH to U/$\overline{D}$ change	100			ns
t_{DI}	U/$\overline{D}$ to $\overline{INC}$ setup	2.9			µs
t_{IL}	$\overline{INC}$ LOW period	1			µs
t_{IH}	$\overline{INC}$ HIGH period	1			µs
t_{IC}	$\overline{INC}$ inactive to $\overline{CS}$ inactive	1			µs
t_{CPH}	$\overline{CS}$ deselect time (STORE)	20			ms
t_{CPH}	$\overline{CS}$ deselect time (NO STORE)	100			ns
t_{IW}	$\overline{INC}$ to $V_{W/RW}$ change		100	500	µs
t_{CYC}	$\overline{INC}$ cycle time	4			µs
t_R, t_F(7)	$\overline{INC}$ input rise and fall time			500	µs
t_{PU}(7)	Power up to wiper stable			500	µs
t_R V_{CC}(7)	V_{CC} power-up rate	0.2		50	V/ms

POWER UP AND DOWN REQUIREMENTS

At all times, voltages on the potentiometer pins must be less than $\pm V_{CC}$. The recall of the wiper position from non-volatile memory is not in effect until the V_{CC} supply reaches its final value. The V_{CC} ramp rate spec is always in effect.

A.C. TIMING



Notes: (6) Typical values are for T_A = 25°C and nominal supply voltage.
(7) This parameter is periodically sampled and not 100% tested.
(8) MI in the A.C. timing diagram refers to the minimum incremental change in the V_W output due to a change in the wiper position.

DETAILED PIN DESCRIPTIONS

R_H/V_H and R_L/V_L

The high (V_H/R_H) and low (V_L/R_L) terminals of the X9C102/103/104/503 are equivalent to the fixed terminals of a mechanical potentiometer. The minimum voltage is –5V and the maximum is +5V. The terminology of V_H/R_H and V_L/R_L references the relative position of the terminal in relation to wiper movement direction selected by the U/$\overline{D}$ input and not the voltage potential on the terminal.

R_W/V_W

V_W/R_W is the wiper terminal, and is equivalent to the movable terminal of a mechanical potentiometer. The position of the wiper within the array is determined by the control inputs. The wiper terminal series resistance is typically 40Ω.

Up/Down (U/$\overline{D}$)

The U/$\overline{D}$ input controls the direction of the wiper movement and whether the counter is incremented or decremented.

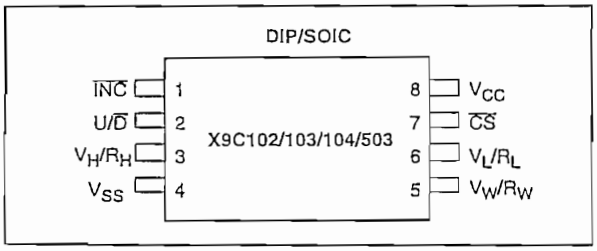
Increment ($\overline{INC}$)

The $\overline{INC}$ input is negative-edge triggered. Toggling $\overline{INC}$ will move the wiper and either increment or decrement the counter in the direction indicated by the logic level on the U/$\overline{D}$ input.

Chip Select ($\overline{CS}$)

The device is selected when the $\overline{CS}$ input is LOW. The current counter value is stored in nonvolatile memory when $\overline{CS}$ is returned HIGH while the $\overline{INC}$ input is also HIGH. After the store operation is complete the X9C102/103/104/503 device will be placed in the low power standby mode until the device is selected once again.

PIN CONFIGURATION



PIN NAMES

Symbol	Description
V_H/R_H	High Terminal
V_W/R_W	Wiper Terminal
V_L/R_L	Low Terminal
V_{SS}	Ground
V_{CC}	Supply Voltage
U/$\overline{D}$	Up/Down Control Input
$\overline{INC}$	Increment Control Input
$\overline{CS}$	Chip Select Control Input
NC	No Connection

PRINCIPLES OF OPERATION

There are three sections of the X9Cxxx: the input control, counter and decode section; the nonvolatile memory; and the resistor array. The input control section operates just like an up/down counter. The output of this counter is decoded to turn on a single electronic switch connecting a point on the resistor array to the wiper output. Under the proper conditions the contents of the counter can be stored in nonvolatile memory and retained for future use. The resistor array is comprised of 99 individual resistors connected in series. At either end of the array and between each resistor is an electronic switch that transfers the potential at that point to the wiper.

The wiper, when at either fixed terminal, acts like its mechanical equivalent and does not move beyond the last position. That is, the counter does not wrap around when clocked to either extreme.

The electronic switches on the device operate in a "make before break" mode when the wiper changes tap positions. If the wiper is moved several positions, multiple taps are connected to the wiper for t_{IW} ($\overline{INC}$ to V_W/R_W change). The R_{TOTAL} value for the device can temporarily be reduced by a significant amount if the wiper is moved several positions.

When the device is powered-down, the last wiper position stored will be maintained in the nonvolatile memory. When power is restored, the contents of the memory are recalled and the wiper is set to the value last stored.

INSTRUCTIONS AND PROGRAMMING

The $\overline{INC}$, U/$\overline{D}$ and $\overline{CS}$ inputs control the movement of the wiper along the resistor array. With $\overline{CS}$ set LOW the device is selected and enabled to respond to the U/$\overline{D}$ and $\overline{INC}$ inputs. HIGH to LOW transitions on $\overline{INC}$ will increment or decrement (depending on the state of the U/$\overline{D}$ input) a seven-bit counter. The output of this counter is decoded to select one of one-hundred wiper positions along the resistive array.

The value of the counter is stored in nonvolatile memory whenever $\overline{CS}$ transitions HIGH while the $\overline{INC}$ input is also HIGH.

The system may select the X9Cxxx, move the wiper, and deselect the device without having to store the latest wiper position in nonvolatile memory. After the wiper movement is performed as described above and once the new position is reached, the system must keep $\overline{INC}$ LOW while taking $\overline{CS}$ HIGH. The new wiper position will be maintained until changed by the system or until a power-down/up cycle recalled the previously stored data.

This procedure allows the system to always power-up to a preset value stored in nonvolatile memory; then during system operation minor adjustments could be made. The adjustments might be based on user preference: system parameter changes due to temperature drift, etc...

The state of U/$\overline{D}$ may be changed while $\overline{CS}$ remains LOW. This allows the host system to enable the device and then move the wiper up and down until the proper trim is attained.

MODE SELECTION

$\overline{CS}$	$\overline{INC}$	U/$\overline{D}$	Mode
L	↘	H	Wiper Up
L	↘	L	Wiper Down
↗	H	X	Store Wiper Position
H	X	X	Standby Current
↗	L	X	No Store, Return to Standby

SYMBOL TABLE

WAVEFORM	INPUTS	OUTPUTS
	Must be steady	Will be steady
	May change from Low to High	Will change from Low to High
	May change from High to Low	Will change from High to Low
	Don't Care: Changes Allowed	Changing: State Not Known
	N/A	Center Line is High Impedance

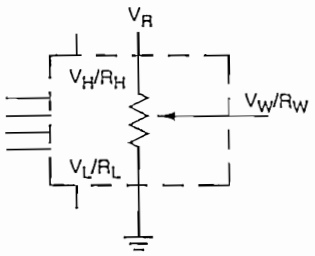
PERFORMANCE CHARACTERISTICS

Contact the factory for more information.

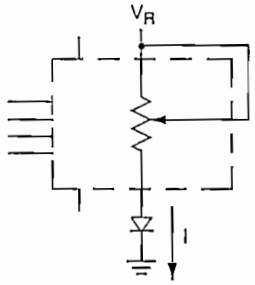
APPLICATIONS INFORMATION

Electronic digitally controlled (XCDP) potentiometers provide three powerful application advantages; (1) the variability and reliability of a solid-state potentiometer, (2) the flexibility of computer-based digital controls, and (3) the retentivity of nonvolatile memory used for the storage of multiple potentiometer settings or data.

Basic Configurations of Electronic Potentiometers



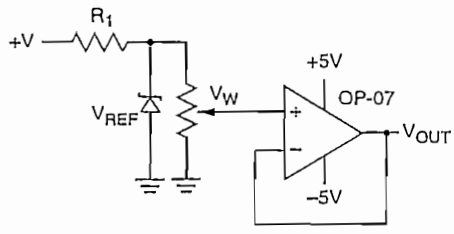
Three terminal potentiometer; variable voltage divider



Two terminal variable resistor; variable current

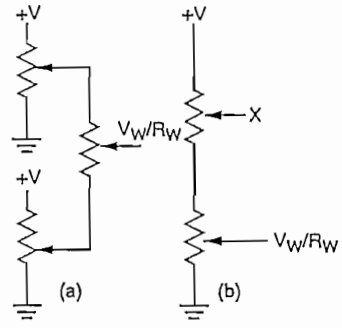
Basic Circuits

Buffered Reference Voltage

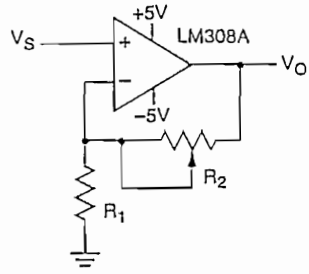


$V_{OUT} = V_W/R_W$

Cascading Techniques

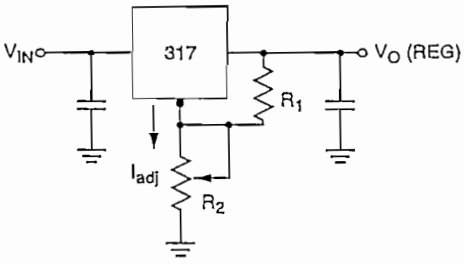


Noninverting Amplifier



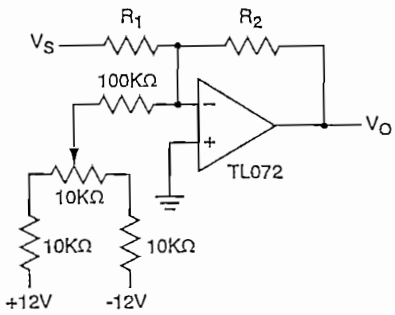
$V_O = (1+R_2/R_1)V_S$

Voltage Regulator

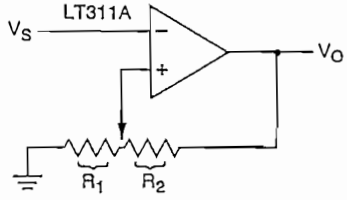


V_O (REG) = 1.25V $(1+R_2/R_1)+I_{adj} R_2$

Offset Voltage Adjustment



Comparator with Hysteresis

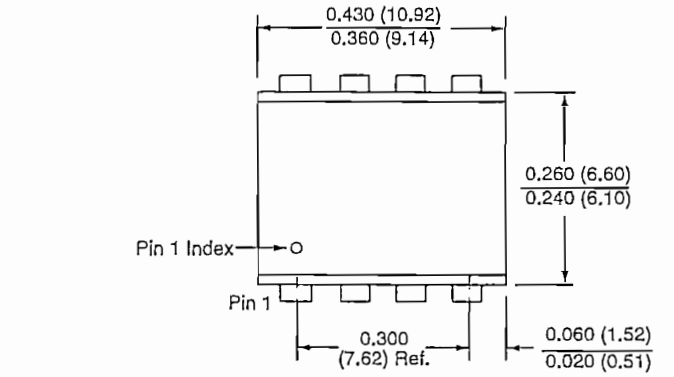


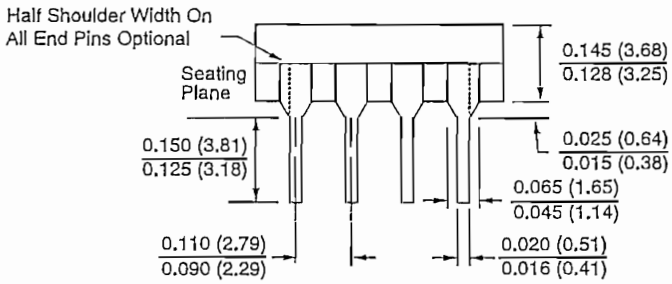
$V_{UL} = \{R_1/(R_1+R_2)\}\ V_O(max)$
$V_{LL} = \{R_1/(R_1+R_2)\}\ V_O(min)$

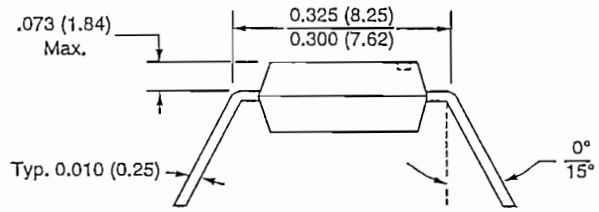
(for additional circuits see AN115)

PACKAGING INFORMATION

8-Lead Plastic Dual In-Line Package Type P



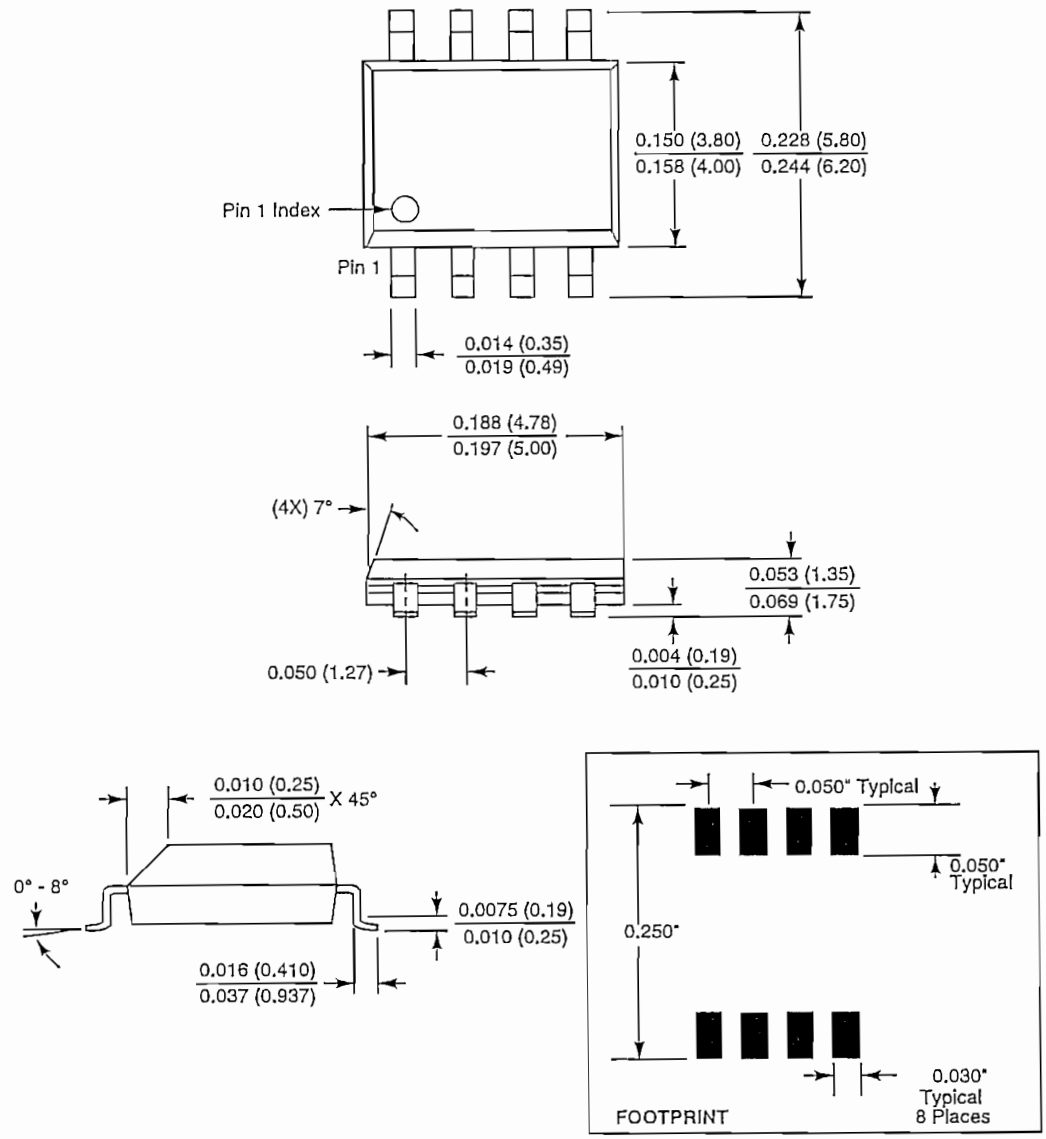




NOTE:
1. ALL DIMENSIONS IN INCHES (IN PARENTHESES IN MILLIMETERS)
2. PACKAGE DIMENSIONS EXCLUDE MOLDING FLASH

PACKAGING INFORMATION

8-Lead Plastic Small Outline Gull Wing Package Type S



NOTE: ALL DIMENSIONS IN INCHES (IN PARENTHESES IN MILLIMETERS)

LIMITED WARRANTY

©Xicor, Inc. 2003 Patents Pending

Devices sold by Xicor, Inc. are covered by the warranty and patent indemnification provisions appearing in its Terms of Sale only. Xicor, Inc. makes no warranty, express, statutory, implied, or by description regarding the information set forth herein or regarding the freedom of the described devices from patent infringement. Xicor, Inc. makes no warranty of merchantability or fitness for any purpose. Xicor, Inc. reserves the right to discontinue production and change specifications and prices at any time and without notice.

Xicor, Inc. assumes no responsibility for the use of any circuitry other than circuitry embodied in a Xicor, Inc. product. No other circuits, patents, or licenses are implied.

TRADEMARK DISCLAIMER:

Xicor and the Xicor logo are registered trademarks of Xicor, Inc. AutoStore, Direct Write, Block Lock, SerialFlash, MPS, BlasLock and XDCP are also trademarks of Xicor, Inc. All others belong to their respective owners.

U.S. PATENTS

Xicor products are covered by one or more of the following U.S. Patents: 4,326,134; 4,393,481; 4,404,475; 4,450,402; 4,486,769; 4,488,060; 4,520,461; 4,533,846; 4,599,706; 4,617,652; 4,668,932; 4,752,912; 4,829,482; 4,874,967; 4,883,976; 4,980,859; 5,012,132; 5,003,197; 5,023,694; 5,084,667; 5,153,880; 5,153,691; 5,161,137; 5,219,774; 5,270,927; 5,324,676; 5,434,396; 5,544,103; 5,587,573; 5,835,409; 5,977,585. Foreign patents and additional patents pending.

LIFE RELATED POLICY

In situations where semiconductor component failure may endanger life, system designers using this product should design the system with appropriate error detection and correction, redundancy and back-up features to prevent such an occurrence.

Xicor's products are not authorized for use in critical components in life support devices or systems.

1. Life support devices or systems are devices or systems which, (a) are intended for surgical implant into the body, or (b) support or sustain life, and whose failure to perform, when properly used in accordance with instructions for use provided in the labeling, can be reasonably expected to result in a significant injury to the user.
2. A critical component is any component of a life support device or system whose failure to perform can be reasonably expected to cause the failure of the life support device or system, or to affect its safety or effectiveness.

Ordering number: EN 1649C

Thick Film Hybrid IC

STK4141V

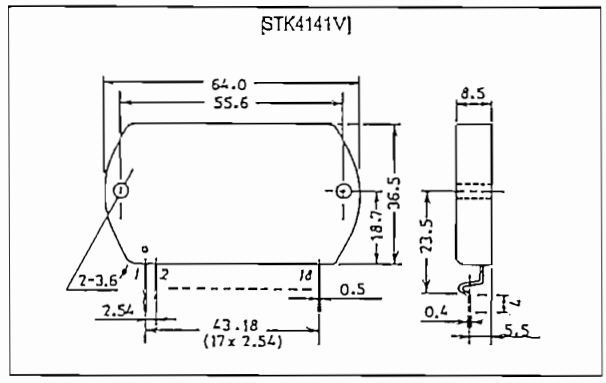
AF Power Amplifier (Split Power Supply) (25W + 25W min, THD = 0.08%)

Features

- The STK4102II series (THD=0.4%), STK4201V series (THD=0.08%) and STK4141X series (THD=0.02%) are pin-compatible. Once the PCB pattern is designed, you can easily satisfy the repuirements for new sets simply by changing the IC.
- Built-in muting circuit to cut off various kinds of pop noise.
- Greatly reduced heat sink due to substrate temperature 125°C guaranteed.
- Current mirror circuit application reduces distortion to 0.08%.

Package Dimensions

unit: mm
4040



Specifications

Maximum Ratings at Ta = 25°C

Parameter	Symbol	Conditions	Ratings	Unit
Maximum supply voltage	V_{CC} max		±40.5	V
Thermal resistance	θj-c		2.6	°C/W
Junction temperature	Tj		150	°C
Operating substrate temperature	Tc		125	°C
Storage temperature	Tstg		–30 to +125	°C
Available time for load short-circuit	t_s	V_{CC} = ±27V, R_L = 8Ω, f = 50Hz, P_O = 25W	2	s

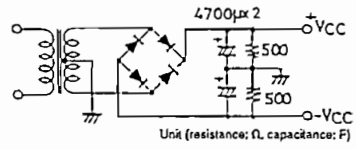
Recommended Operating Conditions at Ta = 25°C

Parameter	Symbol	Conditions	Ratings	Unit
Recommended supply voltage	V_{CC}		±27.0	V
Load resistance	R_L		8	Ω

Operating Characteristics at Ta = 25°C, V_{CC} = ±27.0, R_L = 8Ω(non-inductive), Rg = 600Ω, VG = 40dB

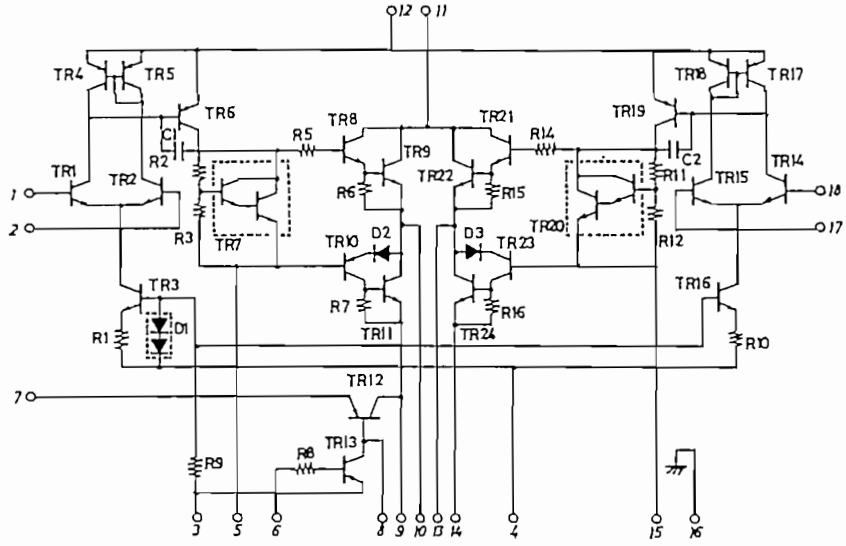
Parameter	Symbol	Conditions	min	typ	max	Unit
Quiescent current	I_{CCO}	V_{CC} = ±32.5V	20	40	100	mA
Output power	P_O1	f = 20Hz to 20kHz, THD = 0.08%	25			W
	P_O2	V_{CC} = ±24V, f = 1kHz, THD = 0.2%, R_L = 4Ω	25			W
Total harmonic distortion	THD	f = 1kHz, P_O = 1W			0.08	%
Frequency response	f_L, f_H	P_O = 1W, $^{+0}_{-3}$ dB		20 to 50k		Hz
Input impedance	r_i	f = 1kHz, P_O = 1W		55		kΩ
Output noise voltage	V_{NO}	V_{CC} = ±32.5V, Rg = 10kΩ			1.2	mVrms
Neutral voltage	V_N	V_{CC} = ±32.5V	–70	0	+70	mV
Muting voltage	V_M		-2	-5	-10	V

Notes. For power supply at the time of test, use a constant-voltage power supply unless otherwise specified.
For measurement of the available time for load short-circuit and output noise voltage, use the specified transformer power supply shown below.
The output noise voltage is represented by the peak value on rms scale (VTVM) of average value indicating type. For AC power supply, use an AC stabilized power supply (50Hz) to eliminate the effect of flicker noise in AC primary line.

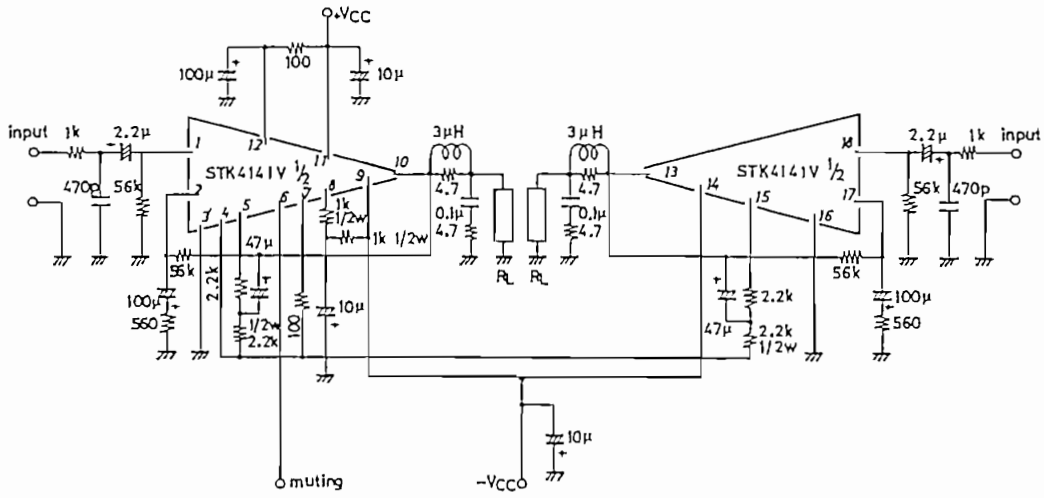


Specified Transformer Power Supply
(Equivalent to RP-25)

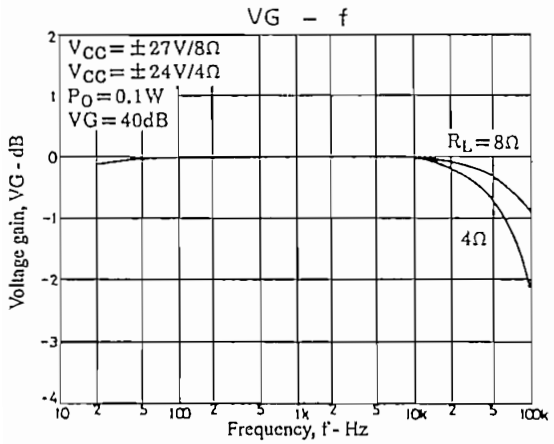
Equivalent Circuit

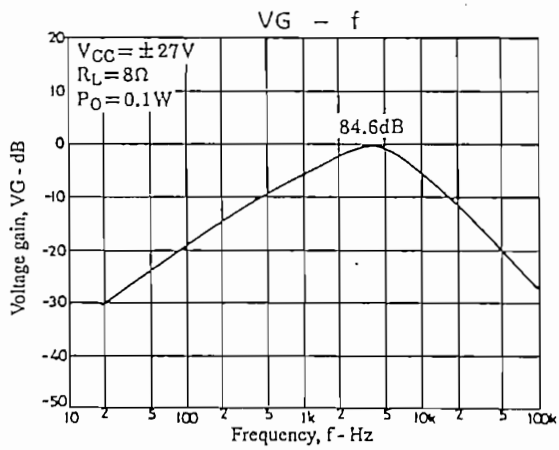


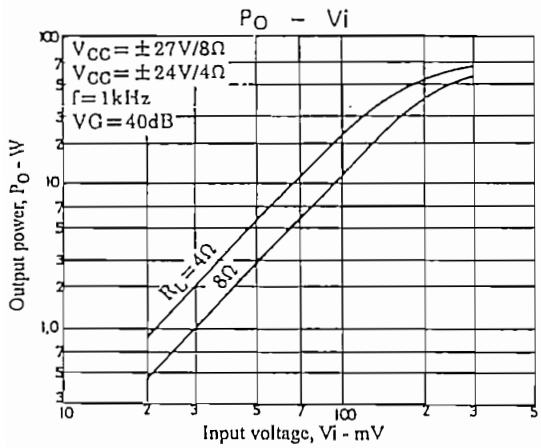
Sample Application Circuit: 25W min AF Power Amplifier (2channels)

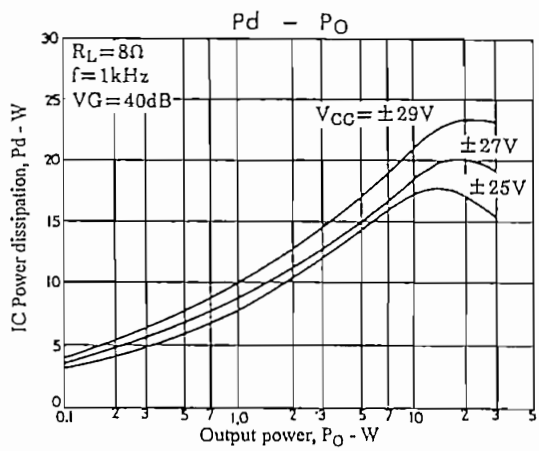


Unit (resistance: Ω, capacitance: F)

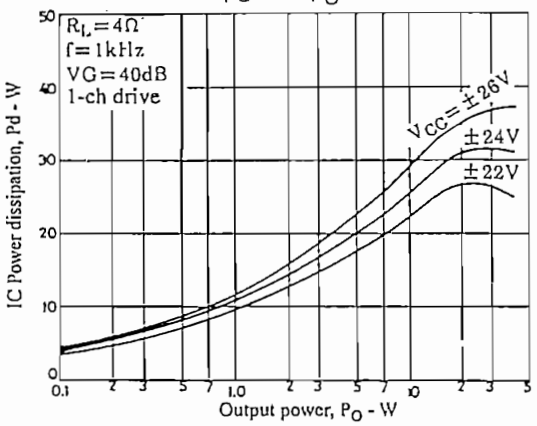




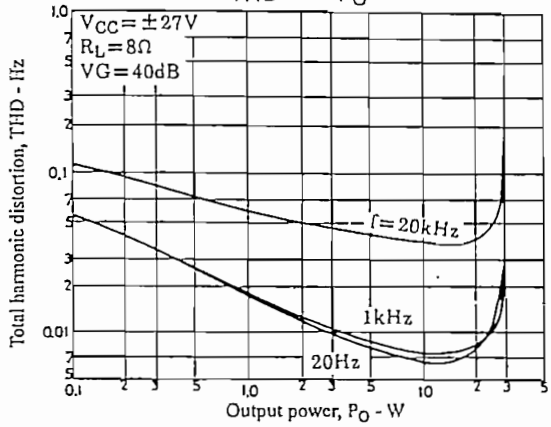




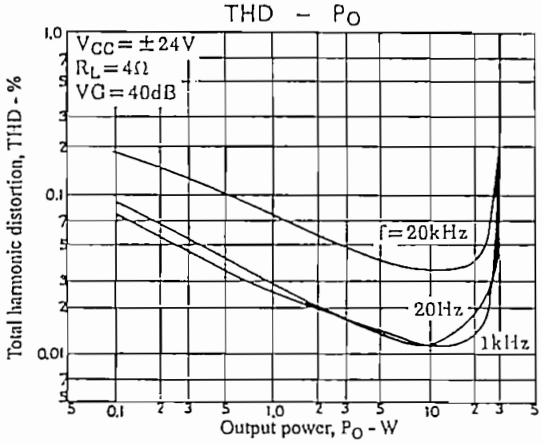
Pd – Po



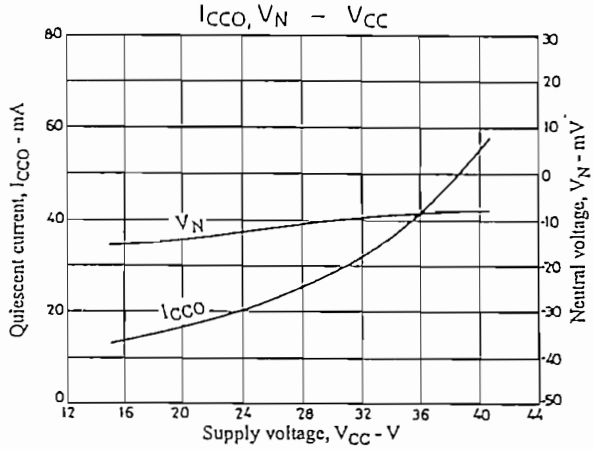
THD – Po



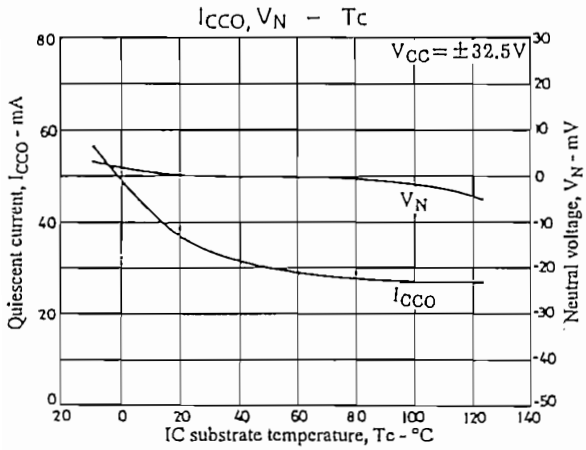
THD – Po



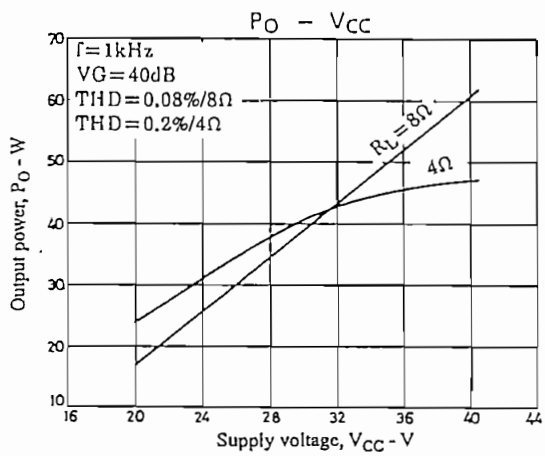
ICCO, VN – VCC

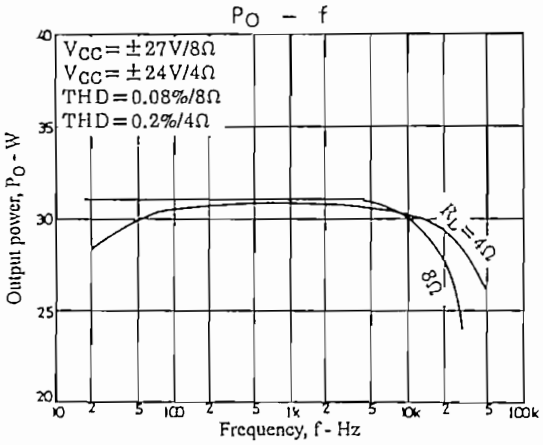


ICCO, VN – Tc



PO – VCC

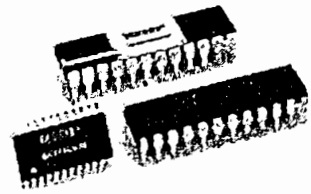




- No products described or contained herein are intended for use in surgical implants, life-support systems, aerospace equipment, nuclear power control systems, vehicles, disaster/crime-prevention equipment and the like, the failure of which may directly or indirectly cause injury, death or property loss.
- Anyone purchasing any products described or contained herein for an above-mentioned use shall:
  - ① Accept full responsibility and indemnify and defend SANYO ELECTRIC CO., LTD., its affiliates, subsidiaries and distributors and all their officers and employees, jointly and severally, against any and all claims and litigation and all damages, cost and expenses associated with such use:
  - ② Not impose any responsibility for any fault or negligence which may be cited in any such claim or litigation on SANYO ELECTRIC CO., LTD., its affiliates, subsidiaries and distributors or any of their officers and employees, jointly or severally.
- Information (including circuit diagrams and circuit parameters) herein is for example only; it is not guaranteed for volume production. SANYO believes information herein is accurate and reliable, but no guarantees are made or implied regarding its use or any infringements of intellectual property rights or other rights of third parties.

This catalog provides information as of June, 1997. Specifications and information herein are subject to change without notice.

BURR-BROWN®



ADS7800

12-Bit 3µs Sampling ANALOG-TO-DIGITAL CONVERTER

FEATURES

- 333k SAMPLES PER SECOND
- STANDARD ±10V AND ±5V INPUT RANGES
- DC PERFORMANCE OVER TEMP:
  No Missing Codes
  1/2LSB Integral Linearity Error
  3/4LSB Differential Linearity Error
- AC PERFORMANCE OVER TEMP:
  72dB Signal-to-Noise Ratio
  80dB Spurious-free Dynamic Range
  –80dB Total Harmonic Distortion
- INTERNAL SAMPLE/HOLD, REFERENCE, CLOCK, AND 3-STATE OUTPUTS
- POWER DISSIPATION: 215mW max
- PACKAGE: 24-Pin Single-wide DIP
  24-Lead SOIC

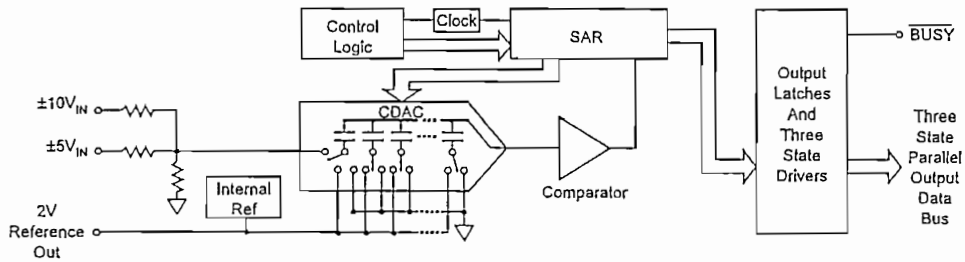
DESCRIPTION

The ADS7800 is a complete 12-bit sampling analog-to-digital converter using state-of-the-art CMOS structures. It contains a complete 12-bit successive approximation A/D converter with internal sample/hold, reference, clock, digital interface for microprocessor control, and three-state output drivers.

The ADS7800 is specified at a 333kHz sampling rate. Conversion time is factory set for 2.70µs max over temperature, and the high speed sampling input stage insures a total acquisition and conversion time of 3µs max over temperature. Precision, laser-trimmed scaling resistors provide industry-standard input ranges of ±5V or ±10V.

AC and DC performance are completely specified. Two grades based on linearity and dynamic performance are available to provide the optimum price/performance fit in a wide range of applications.

The 24-pin ADS7800 is available in plastic and side-braze hermetic 0.3" wide DIPs, and in an SOIC package. It operates from a +5V supply and either a –12V or –15V supply. The ADS7800 is available in grades specified over 0°C to +70°C and –40°C to +85°C temperature ranges.



International Airport Industrial Park • Mailing Address: PO Box 11400, Tucson, AZ 85734 • Street Address: 6730 S. Tucson Blvd., Tucson, AZ 85706 • Tel: (520) 746-1111 • Twx: 910-952-1111
Internet: http://www.burr-brown.com/ • FAXLine: (800) 548-6133 (US/Canada Only) • Cable: BBRCORP • Telex: 066-6491 • FAX: (520) 889-1510 • Immediate Product Info: (800) 548-6132

© 1989 Burr-Brown Corporation

PDS-1018E

Printed in U.S.A. October, 1993

SPECIFICATIONS

ELECTRICAL

At $T_A = T_{MIN}$ to T_{MAX}, Sampling Frequency, f_S, = 333kHz, $-V_S = -15V$, $V_S = +5V$, unless otherwise specified.

PARAMETER	CONDITIONS	ADS7800JP/JU/AH			ADS7800KP/KU/BH			UNITS
		MIN	TYP	MAX	MIN	TYP	MAX	
RESOLUTION				12			"	Bits
ANALOG INPUT								
Voltage Ranges			±10V/±5V			"		V
Impedance	±10V Range	4.4	6.3	8.1	"	"	"	kΩ
	±5V Range	2.9	4.2	5.4	"	"	"	kΩ
THROUGHPUT SPEED								
Conversion Time	Conversion Alone		2.5	2.7		"	"	µs
Complete Cycle	Acquisition + Conversion		2.6	3.0		"	"	µs
Throughput Rate		333	380		"	"		kHz
DC ACCURACY								
Full Scale Error[1]				±0.50			±0.35	%
Full Scale Error Drift			6			"		ppm/°C
Integral Linearity Error				±1			±1/2	LSB[2]
Differential Linearity Error				±1			±3/4	LSB
No Missing Codes			Guaranteed			Guaranteed		
Bipolar Zero[1]				±4			±2	LSB
Bipolar Zero Drift			1			"		ppm/°C
Power Supply Sensitivity								
	$-16.5V < -V_S < -13.5V$			±1/2			"	LSB
	$-12.6V < -V_S < -11.4V$			±1/2			"	LSB
	$+4.75V < V_S < +5.25V$			±1			±1/2	LSB
Transition Noise[3]			0.1			"		LSB
AC ACCURACY								
Spurious-Free Dynamic Range	f_{IN} = 47kHz	74	77		77	80		dB[4]
Total Harmonic Distortion	f_{IN} = 47kHz		–77	–74		–80	–77	dB
Two-tone Intermodulation Distortion	f_{IN1} = 24.4kHz (–6dB) f_{IN2} = 28.5kHz (–6dB)		–77	–74		–80	–77	dB
Signal-to-(Noise + Distortion) Ratio	f_{IN} = 47kHz	67	70		69	72		dB
Signal-to-Noise Ratio (SNR)	f_{IN} = 47kHz	68	71		70	73		dB
SAMPLING DYNAMICS								
Aperture Delay			13			"		ns
Aperture Jitter			150			"		ps, rms
Transient Response[5]			130			"		ns
Overvoltage Recovery[6]			150			"		ns
INTERNAL REFERENCE VOLTAGE								
Voltage		1.9	2.0	2.1	"	"	"	V
Source Current Available for External Loads			10			"		µA
DIGITAL INPUTS								
Logic Levels								
V_{IL}		–0.3		+0.8	"		"	V
V_{IH}		+2.4		+5.3	"		"	V
I_{IL}		–5			"			µA
I_{IH}		+5			"			µA
DIGITAL OUTPUTS								
Data Format				Parallel, 12-bit or 8-bit/4-bit				
Data Coding				Binary Offset Binary				
V_{OL}	I_{SINK} = 1.6mA	0.0		+0.4	"		"	V
V_{OH}	I_{SOURCE} = 500µA	+2.4		+5.0	"		"	V
$I_{LEAKAGE}$ (High-Z State)			±0.1	±5		"	"	µA
POWER SUPPLIES								
Rated Voltage								
$-V_S$		–11.4	–15	–16.5	"	"	"	V
V_S (V_{SA} and V_{SD})		+4.75	+5.0	+5.25	"	"	"	V
Current								
$-I_S$			3.5	6		"	"	mA
I_S			18	25		"	"	mA
Power Consumption			135	215		"	"	mW

SPECIFICATIONS (CONT)

ELECTRICAL

At $T_A = T_{MIN}$ to T_{MAX}, Sampling Frequency, f_S, = 333kHz, $-V_S = -15V$, $V_S = +5V$, unless otherwise specified.

PARAMETER	CONDITIONS	ADS7800JP/JU/AH			ADS7800KP/KU/BH			UNITS
		MIN	TYP	MAX	MIN	TYP	MAX	
TEMPERATURE RANGE								
Specification	JP/JU/KP/KU	0		+70	*		*	°C
	AH/BH	–40		+85	*		*	°C
Operating	JP/KP/JU/KU	–40		+85	*		*	°C
Storage		–65		+150	*		*	°C

* Same as specification for ADS7800JP/JU/AH.

NOTES: (1) Adjustable to zero with external potentiometer. (2) LSB means Least Significant Bit. For ADS7800, 1LSB = 2.44mV for the ±5V range, 1LSB = 4.88mV for the ±10V range. (3) Noise was characterized over temperature near full scale, 0V, and negative full scale. 0.1LSB represents a typical rms level of noise at the worst case, which was near full scale input at +125°C. (4) All specifications in dB are referred to a full-scale input, either ±10V or ±5V. (5) For full scale step input, 12-bit accuracy attained in specified time. (6) Recovers to specified performance in specified time after 2 x F_S input overvoltage.

ABSOLUTE MAXIMUM RATINGS

$-V_S$ to ANALOG COMMON	–16.5V
V_S to DIGITAL COMMON	+7V
Pin 23 (V_{SD}) to Pin 24 (V_{SA})	±0.3V
ANALOG COMMON to DIGITAL COMMON	±1V
Control Inputs to DIGITAL COMMON	–0.3 to V_S + 0.3V
Analog Input Voltage	±20V
Maximum Junction Temperature	160°C
Internal Power Dissipation	750mW
Lead Temperature (soldering, 10s)	+300°C
Thermal Resistance, θ_{JA}:	
Plastic DIP	100°C/W
SOIC	100°C/W
Ceramic	50°C/W

ELECTROSTATIC DISCHARGE SENSITIVITY

The ADS7800 is an ESD (electrostatic discharge) sensitive device. The digital control inputs have a special FET structure, which turns on when the input exceeds the supply by 18V, to minimize ESD damage. However, permanent damage may occur on unconnected devices subject to high energy electrostatic fields. When not in use, devices must be stored in conductive foam or shunts. The protective foam should be discharged to the destination socket before devices are removed.

PACKAGE/ORDERING INFORMATION

PRODUCT	INTEGRAL LINEARITY ERROR (LSB)	SIGNAL-TO-(NOISE+DISTORTION) RATIO (dB min)	SPECIFICATION TEMPERATURE RANGE (°C)	PACKAGE	PACKAGE DRAWING NUMBER[1]
ADS7800JP	±1	67	0 to +70	24-Pin Plastic DIP	243
ADS7800KP	±1/2	69	0 to +70	24-Pin Plastic DIP	243
ADS7800JU	±1	67	0 to +70	24-Pin Plastic SOIC	239
ADS7800KU	±1/2	69	0 to +70	24-Pin Plastic SOIC	239
ADS7800AH	±1	67	–40 to +85	24-Pin Ceramic DIP	245
ADS7800BH	±1/2	69	–40 to +85	24-Pin Ceramic DIP	245

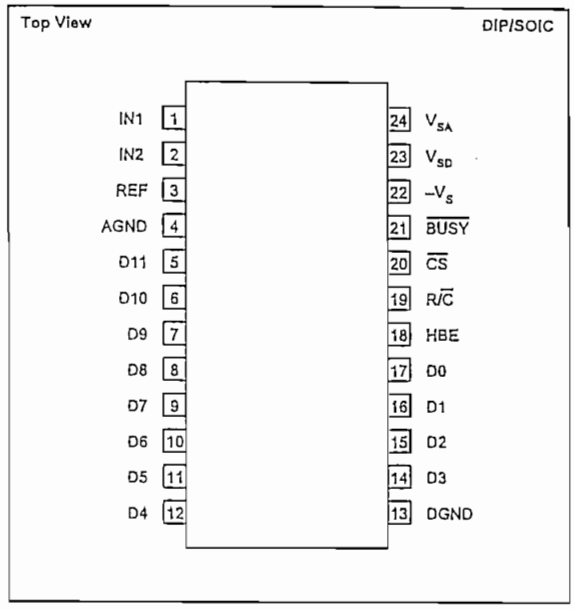
NOTE: (1) For detailed drawing and dimension table, please see end of data sheet, or Appendix C of Burr-Brown IC Data Book.

The information provided herein is believed to be reliable; however, BURR-BROWN assumes no responsibility for inaccuracies or omissions. BURR-BROWN assumes no responsibility for the use of this information, and all use of such information shall be entirely at the user's own risk. Prices and specifications are subject to change without notice. No patent rights or licenses to any of the circuits described herein are implied or granted to any third party. BURR-BROWN does not authorize or warrant any BURR-BROWN product for use in life support devices and/or systems.

PIN ASSIGNMENTS

PIN #	NAME	DESCRIPTION
1	IN1	±10V Analog Input. Connected to GND for ±5V range.
2	IN2	±5V Analog Input. Connected to GND for ±10V range.
3	REF	+2V Reference Output. Bypass to GND with 22µF to 47µF Tantalum. Buffer for external loads.
4	AGND	Analog Ground. Connect to pin 13.
5	D11	Data Bit 11. Most Significant Bit (MSB).
6	D10	Data Bit 10.
7	D9	Data Bit 9.
8	D8	Data Bit 8.
9	D7	Data Bit 7 if HBE is LOW; LOW if HBE is HIGH.
10	D6	Data Bit 6 if HBE is LOW; LOW if HBE is HIGH.
11	D5	Data Bit 5 if HBE is LOW; LOW if HBE is HIGH.
12	D4	Data Bit 4 if HBE is LOW; LOW if HBE is HIGH.
13	DGND	Digital Ground. Connect to pin 4.
14	D3	Data Bit 3 if HBE is LOW; Data Bit 11 if HBE is HIGH.
15	D2	Data Bit 2 if HBE is LOW; Data Bit 10 if HBE is HIGH.
16	D1	Data Bit 1 if HBE is LOW; Data Bit 9 if HBE is HIGH.
17	D0	Data Bit 0 if HBE is LOW. Least Significant Bit (LSB); Data Bit 8 if HBE is HIGH.
18	HBE	High Byte Enable. When held LOW, data output as 12-bits in parallel. When held HIGH, four MSBs presented on pins 14-17, pins 9-12 output LOWs. Must be LOW to initiate conversion.
19	R/$\overline{C}$	Read/Convert. Falling edge initiates conversion when $\overline{CS}$ is LOW, HBE is LOW, and $\overline{BUSY}$ is HIGH.
20	$\overline{CS}$	Chip Select. Outputs in Hi-Z state when HIGH. Must be LOW to initiate conversion or read data.
21	$\overline{BUSY}$	Busy. Output LOW during conversion. Data valid on rising edge in Convert Mode.
22	$-V_S$	Negative Power Supply. –12V or –15V. Bypass to GND.
23	V_{SD}	Positive Digital Power Supply. +5V. Connect to pin 24, and bypass to GND.
24	V_{SA}	Positive Analog Power Supply. +5V. Connect to pin 23, and bypass to GND.

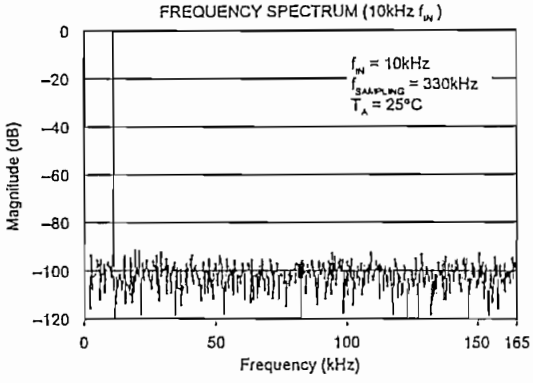
PIN CONFIGURATION



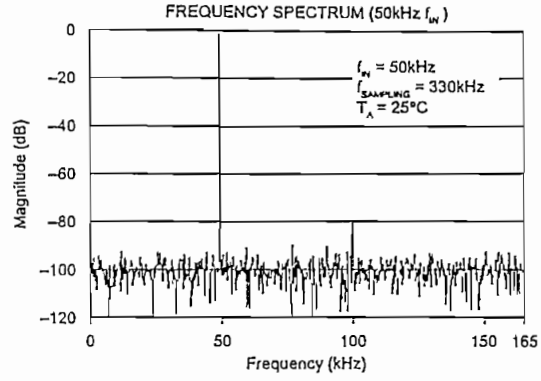
TYPICAL PERFORMANCE CURVES

At $+V_S$ = +5V, $-V_S$ = –15V, and T_A = +25°C, unless otherwise noted. All plots use 1024 point FFTs.

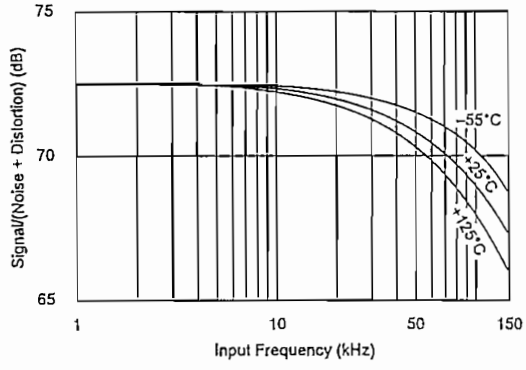
FREQUENCY SPECTRUM (10kHz f_{IN})



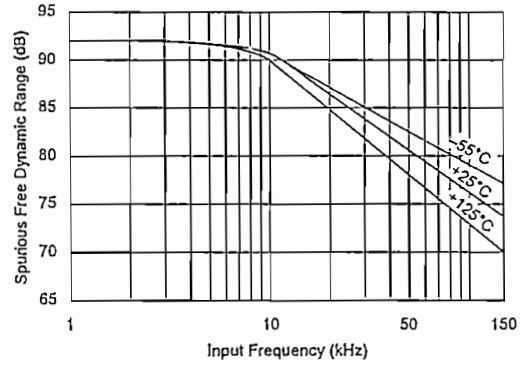
FREQUENCY SPECTRUM (50kHz f_{IN})



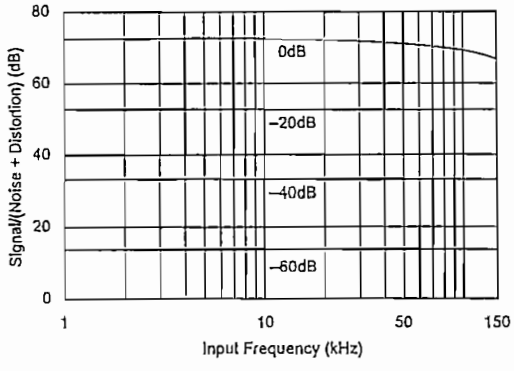
SIGNAL/(NOISE + DISTORTION) vs
INPUT FREQUENCY AND AMBIENT TEMPERATURE



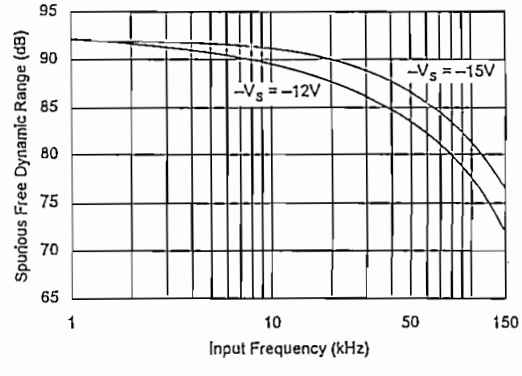
SPURIOUS FREE DYNAMIC RANGE vs
INPUT FREQUENCY AND AMBIENT TEMPERATURE



SIGNAL/(NOISE + DISTORTION) vs
FREQUENCY AND AMPLITUDE



SPURIOUS FREE DYNAMIC RANGE vs
INPUT FREQUENCY AND NEGATIVE SUPPLY VOLTAGE



THEORY OF OPERATION

The ADS7800 combines the advantages of advanced CMOS technology (logic density, stable capacitors, and good analog switches) with Burr-Brown's proven skills in laser-trimmed thin-film resistors to provide a complete sampling A/D converter.

A basic charge-redistribution successive approximation architecture converts analog input voltages into digital words. Figure 1 shows the operation of a simplified three bit charge redistribution A/D. Precision laser-trimmed scaling resistors at the input divide standard input ranges (±10V or ±5V for the ADS7800) into levels compatible with the CMOS characteristics of the internal capacitor array.

While in the sampling mode, the capacitor array switch for the MSB capacitor (S_1) is in position "S", so that the charge on the MSB capacitor is proportional to the voltage level of the analog input signal, and the remaining array switches (S_2 and S_3) are set to position "R" to provide an accurate bipolar offset from the reference source REF. At the same time, switch S_C is also in the closed position to auto-zero any offset errors in the CMOS comparator.

When a convert command is received, switch S_1 is opened to trap a charge on the MSB capacitor proportional to the input level at the time of the sampling command, switches S_2 and S_3 are opened to trap an offset charge, and switch S_C is opened to float the comparator input. The charge trapped on the capacitor array can now be moved between the three capacitors in the array by connecting switches S_1, S_2 and S_3 to positions "R" (to connect to REF) or "G" (to connect to GND) successively, changing the voltage generated at the comparator input node.

The first approximation connects the MSB capacitor via switch S_1 to REF, while switches S_2 and S_3 are connected to GND. Depending on whether the comparator output is HIGH or LOW, the logic will then latch S_1 in position "R" or "G", and moves on to make the next approximation by connecting S_2 to REF and S_3 to GND. When the three successive approximation steps are made for this simple converter, the voltage level at the comparator will be within 1/2LSB of GND, and the data output word will be based on reading the positions of S_1, S_2 and S_3.

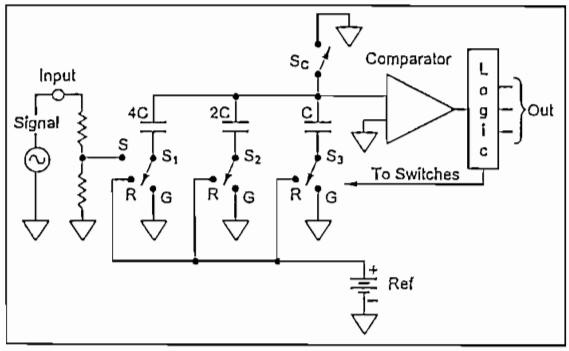


FIGURE 1. 3-Bit Charge Redistribution A/D.

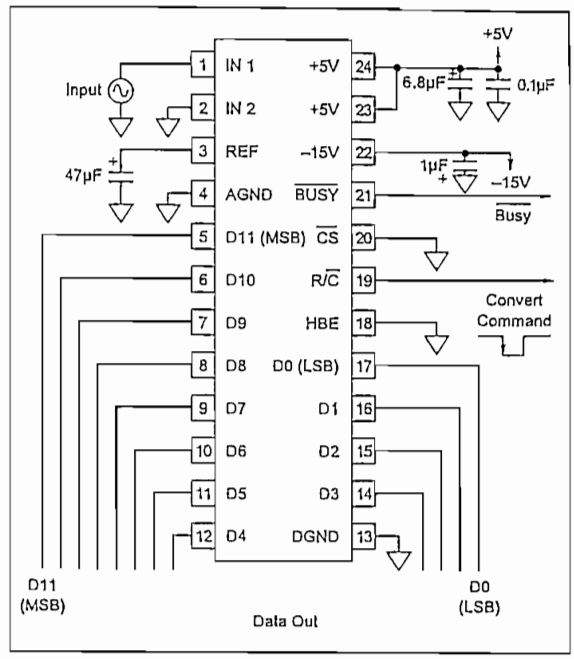


FIGURE 2. Basic ±10V Operation.

OPERATION

BASIC OPERATION

Figure 2 shows the simple hookup circuit required to operate the ADS7800 in a ±10V range in the Convert Mode. A convert command arriving on pin 19, R/$\overline{C}$, (a pulse taking pin 19 LOW for a minimum of 40ns) puts the ADS7800 in the hold mode, and a conversion is started. Pin 21, $\overline{BUSY}$, will be held LOW during the conversion, and rises only after the conversion is completed and the data has been transferred to the output latches. Thus, the rising edge of the signal on pin 21 can be used to read the data from the conversion. Also, during conversion, the $\overline{BUSY}$ signal puts the output data lines in Hi-Z states and inhibits input lines. This means that pulses on pin 19 are ignored, so that new conversions cannot be initiated during a conversion, either as a result of spurious signals or to short-cycle the ADS7800.

In the Read Mode, the input to pin 19 is kept normally LOW, and a HIGH pulse is used to read data and initiate a conversion. In this mode, the rising edge of R/$\overline{C}$ on pin 19 will enable the output data pins, and the data from the previous conversion becomes valid. The falling edge then puts the ADS7800 in a hold mode, and initiates a new conversion.

The ADS7800 will begin acquiring a new sample as soon as the conversion is completed, even before the $\overline{BUSY}$ output rises on pin 21, and will track the input signal until the next conversion is started, whether in the Convert Mode or the Read Mode.

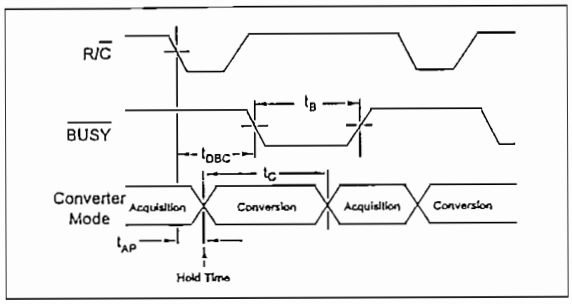


FIGURE 3. Acquisition and Conversion Timing.

SYMBOL	PARAMETER	MIN	TYP	MAX	UNITS
t_{DBC}	$\overline{BUSY}$ delay from R/$\overline{C}$		80	150	ns
t_B	$\overline{BUSY}$ Low		2.5	2.7	µs
t_{AP}	Aperture Delay		13		ns
Δt_{AP}	Aperture Jitter		150		ps, rms
t_C	Conversion Time		2.47	2.70	µs

TABLE I. Acquisition and Conversion Timing.

For use with an 8-bit bus, the data can be read out in two bytes under the control of pin 18, HBE. With a LOW input on pin 18, at the end of a conversion, the 8 LSBs of data are loaded into the latches on pins 9 through 12 and 14 through 17. Taking pin 18 HIGH then loads the 4 MSBs on pins 14 through 17, with pins 9 through 12 being forced LOW.

ANALOG INPUT RANGES

The ADS7800 offers two standard bipolar input ranges: ±10V and ±5V. If a ±10V range is required, the analog input signal should be connected to pin 1. A signal requiring a ±5V range should be connected to pin 2. In either case, the other pin of the two must be grounded or connected to the adjustment circuits described in the section on calibration. (See Figures 4 and 5, or 10 and 11.)

CONTROLLING THE ADS7800

The ADS7800 can be easily interfaced to most microprocessor-based and other digital systems. The microprocessor may take full control of each conversion, or the ADS7800 may operate in a stand-alone mode, controlled only by the R/$\overline{C}$ input. Full control consists of initiating the conversion and reading the output data at user command, transmitting data either all 12-bits in one parallel word, or in two 8-bit bytes. The three control inputs ($\overline{CS}$, R/$\overline{C}$ and HBE) are all TTL/CMOS compatible. The functions of the control lines are shown in Table II.

$\overline{CS}$	R/$\overline{C}$	HBE	$\overline{BUSY}$	OPERATION
1	X	X	1	None - Outputs in Hi-Z State.
0	↓	0	1	Holds Signal and Initiates Conversion.
0	1	0	1	Output Three-State Buffers Enabled once Conversion has Finished.
0	1	1	1	Enable Hi-Byte in 8-bit Bus Mode.
0	↓	1	1	Inhibit Start of Conversion.
0	0	1	1	None - Outputs In Hi-Z State.
X	X	X	0	Conversion in Progress. Outputs Hi-Z State. New Conversion Inhibited until Present Conversion has Finished.

TABLE II. Control Line Functions.

For stand-alone operation, control of the ADS7800 is accomplished by a single control line connected to R/$\overline{C}$. In this mode, $\overline{CS}$ and HBE are connected to GND. The output data are presented as 12-bit words. The stand-alone mode is used in systems containing dedicated input ports which do not require full bus interface capability.

Conversion is initiated by a HIGH-to-LOW transition on R/$\overline{C}$. The three-state data output buffers are enabled when R/$\overline{C}$ is HIGH and $\overline{BUSY}$ is HIGH. Thus, there are two possible modes of operation: conversion can be initiated with either positive or negative pulses. In either case, the R/$\overline{C}$ pulse must remain LOW a minimum of 40ns.

Figure 6 illustrates timing when conversion is initiated by an R/$\overline{C}$ pulse which goes LOW and returns HIGH during the conversion. In this case (Convert Mode), the three-state outputs go into the Hi-Z state in response to the falling edge of R/$\overline{C}$, and are enabled for external access of the data after completion of the conversion.

Figure 7 illustrates the timing when conversion is initiated by a positive R/$\overline{C}$ pulse. In this mode (Read Mode), the output data from the previous conversion is enabled during the HIGH portion of R/$\overline{C}$. A new conversion starts on the falling edge of R/$\overline{C}$, and the three-state outputs return to the Hi-Z state until the next occurrence of a HIGH on R/$\overline{C}$.

CONVERSION START

A conversion is initiated on the ADS7800 only by a negative transition occurring on R/$\overline{C}$, as shown in Table I. No other combination of states or transitions will initiate a conversion. Conversion is inhibited if either $\overline{CS}$ or HBE are HIGH, or if $\overline{BUSY}$ is LOW. $\overline{CS}$ and HBE should be stable a minimum of 25ns prior to the transition on R/$\overline{C}$. Timing relationships for start of conversion are illustrated in Figure 8.

The $\overline{BUSY}$ output indicates the current state of the converter by being LOW only during conversion. During this time the three-state output buffers remain in a Hi-Z state, and therefore data cannot be read during conversion. During this period, additional transitions on the three digital inputs ($\overline{CS}$, R/$\overline{C}$ and HBE) will be ignored, so that conversion cannot be prematurely terminated or restarted.

INTERNAL CLOCK

The ADS7800 has an internal clock that is factory trimmed to achieve a typical conversion time of 2.47µs, and a maximum conversion time over the full operating temperature range of 2.7µs. No external adjustments are required, and with the guaranteed maximum acquisition time of 300ns, throughput performance is assured with convert pulses as close as 3µs.

READING DATA

After conversion is initiated, the output buffers remain in a Hi-Z state until the following three logic conditions are simultaneously met: R/$\overline{C}$ is HIGH, $\overline{BUSY}$ is HIGH and $\overline{CS}$ is LOW. Upon satisfaction of these conditions, the data lines are enabled according to the state of HBE. See Figure 9 and Table III for timing relationships and specifications.

CALIBRATION

OPTIONAL EXTERNAL GAIN AND OFFSET TRIM

Offset and full-scale errors may be trimmed to zero using external offset and full-scale trim potentiometers connected to the ADS7800 as shown in Figures 10 and 11.

If adjustment of offset and full scale is not required, connections as shown in Figures 4 and 5 should be used.

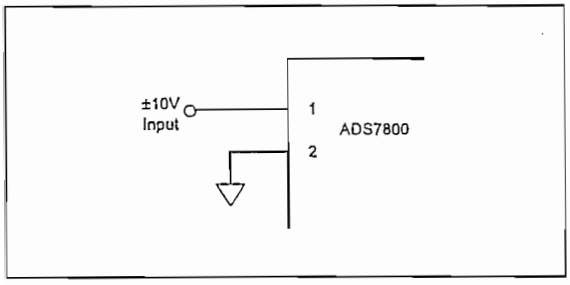


FIGURE 4. ±10V Range Without Trims.

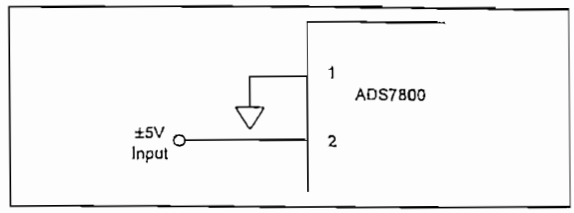


FIGURE 5. ±5V Range Without Trims.

CALIBRATION PROCEDURE

First, trim offset, by applying at the input (pin 1 or 2) the mid-point transition voltage (–2.44mV for the ±10V range, –1.22mV for the ±5V range.) With the ADS7800 converting continually, adjust potentiometer R_1 until the MSB (D11 on pin 5) is toggling alternately HIGH and LOW.

Next adjust full scale, by applying at the input a DC input signal that is 3/2LSB below the nominal full scale voltage (+9.9927V for the ±10V range, +4.9963V for the ±5V range.) With the ADS7800 converting continually, adjust R_2 until the LSB (D0 on pin 17) is toggling HIGH and LOW with all of the other bits HIGH.

LAYOUT CONSIDERATIONS

Because of the high resolution and linearity of the ADS7800, system design problems such as ground path resistance and contact resistance become very important.

ANALOG SIGNAL SOURCE IMPEDANCE

The input resistance of the ADS7800 is 6.3kΩ or 4.2kΩ (for the ±10V and ±5V ranges respectively.) To avoid introducing distortion, the source resistance must be very low, or constant with signal level. The output impedance provided by most op amps is ideal.

Pins 23 (V_{SD}) and 24 (V_{SA}) are not connected internally on the ADS7800, to maximize accuracy on the chip. They should be connected together as close as possible to the unit.

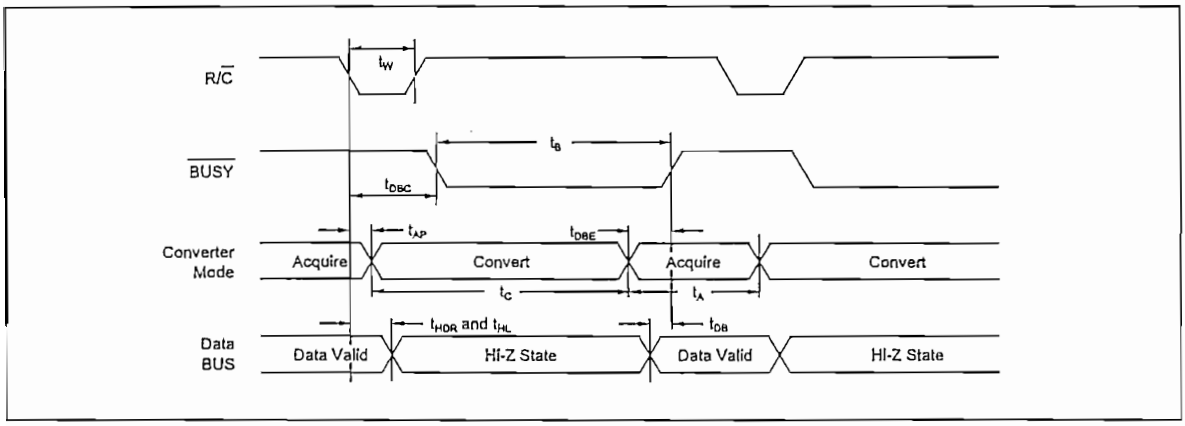


FIGURE 6. Convert Mode: R/$\overline{C}$ Pulse LOW — Outputs Enabled After Conversion.

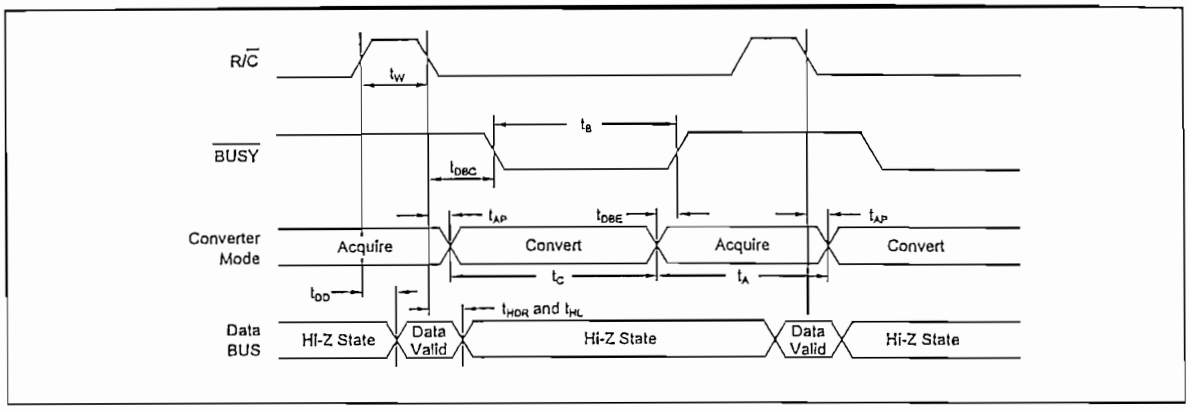


FIGURE 7. Read Mode: R/$\overline{C}$ Pulse HIGH— Outputs Enabled Only When R/$\overline{C}$ is High.

SYMBOL	PARAMETER	MIN	TYP	MAX	UNITS
t_W	R/$\overline{C}$ Pulse Width	40	10		ns
t_{DBC}	$\overline{BUSY}$ delay from R/$\overline{C}$		80	150	ns
t_B	$\overline{BUSY}$ LOW		2.5	2.7	µs
t_{AP}	Aperture Delay		13		ns
Δt_{AP}	Aperture Jitter		150		ps, rms
t_C	Conversion Time		2.47	2.70	µs
t_{DBE}	$\overline{BUSY}$ from End of Conversion		100		ns
t_{DB}	$\overline{BUSY}$ Delay after Data Valid	25	75	200	ns
t_A	Acquisition Time		130	300	ns
t_A+t_C	Throughput Time		2.6	3.0	µs
t_{HDR}	Valid Data Held After R/$\overline{C}$ LOW	20	50		ns
t_S	$\overline{CS}$ or HBE LOW before R/$\overline{C}$ Falls	25	5		ns
t_H	$\overline{CS}$ or HBE LOW after R/$\overline{C}$ Falls	25	0		ns
t_{DD}	Data Valid from $\overline{CS}$ LOW, R/$\overline{C}$ HIGH, and HBE in Desired State (Load = 100pF)		65	150	ns
t_{HDR}	Valid Data Held After R/$\overline{C}$ Low	20	50		ns
t_{HL}	Delay to Hi-Z State after R/$\overline{C}$ Falls or $\overline{CS}$ Rises (3kΩ Pullup or Pulldown)		50	150	ns

TABLE III. Timing Specifications (T_{MIN} to T_{MAX}).

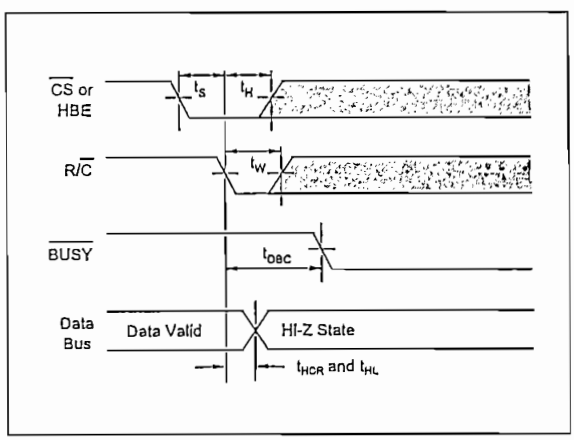


FIGURE 8. Conversion Start Timing.

Pin 24 may be slightly more sensitive than pin 23 to supply variations, but to maintain maximum system accuracy, both should be well isolated from digital supplies with wide load variations.

To limit the effects of digital switching elsewhere in a system on the analog performance of the system, it often makes sense to run a separate +5V supply conductor from the supply regulator to any analog components requiring +5V, including the ADS7800.

The V_S pins (23 and 24) should be connected together and bypassed with a parallel combination of a 6.8µF tantalum capacitor and a 0.1µF ceramic capacitor located close to the converter to obtain noise-free operation. (See Figure 2.) The $-V_S$ pin 22 should be bypassed with a 1µF tantalum capacitor, again as close as possible to the ADS7800.

Noise on the power supply lines can degrade converter performance, especially noise and spikes from a switching power supply. Appropriate supplies or filters must be used.

The GND pins (4 and 13) are also separated internally, and should be directly connected to a ground plane under the

converter if at all possible. A ground plane is usually the best solution for preserving dynamic performance and reducing noise coupling into sensitive converter circuits. Where any compromises must be made, the common return of the analog input signal should be referenced to pin 4, AGND, on the ADS7800, which prevents any voltage drops that might occur in the power supply common returns from appearing in series with the input signal.

Coupling between analog input and digital lines should be minimized by careful layout. For instance, if the lines must cross, they should do so at right angles. Parallel analog and digital lines should be separated from each other by a pattern connected to common.

If external full scale and offset potentiometers are used, the potentiometers and related resistors should be located as close to the ADS7800 as possible.

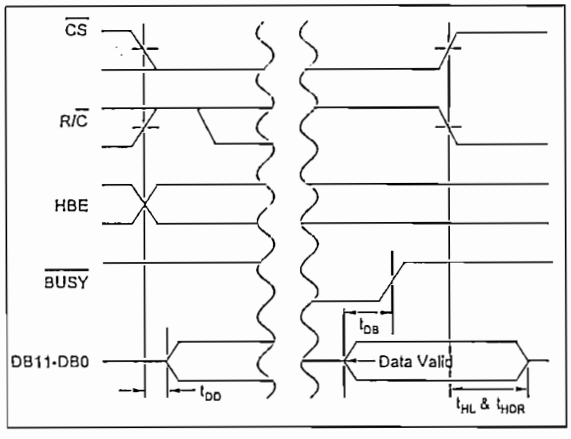


FIGURE 9. Read Cycle Timing.

REFERENCE BYPASS

Pin 3 (REF) should be bypassed with a 22µF to 47µF tantalum capacitor. A rated working voltage of 2V or more is acceptable here. This pin is used to enhance the system accuracy of the internal reference circuit, and is not recommended for driving external signals. If there are important system reasons for using the ADS7800 reference externally, the output of pin 3 must be appropriately buffered.

"HOT SOCKET" PRECAUTION

Two separate +5V V_S pins, 23 and 24, are used to minimize noise caused by digital transients. If one pin is powered and the other is not, the ADS7800 may "Latch Up" and draw excessive current. In normal operation, this is not a problem because both pins will be soldered together. However, during evaluation, incoming inspection, repair, etc., where the potential of a "Hot Socket" exists, care should be taken to power the ADS7800 only after it has been socketed.

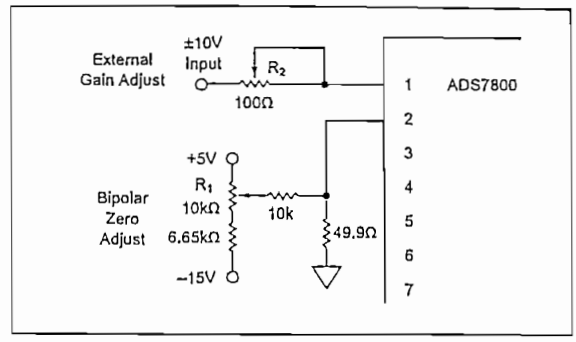


FIGURE 10. ±10V Range With External Trims.

MINIMIZING "GLITCHES"

Coupling of external transients into an A/D converter can cause errors which are difficult to debug. In addition to the discussions earlier on layout considerations for supplies, bypassing and grounding, there are several other useful steps that can be taken to get the best analog performance out of a system using the ADS7800. These potential system problem sources are particularly important to consider when developing a new system, and looking for the causes of errors in breadboards.

First, care should be taken to avoid glitches during critical times in the sampling and conversion process. Since the ADS7800 has an internal sample/hold function, the signal that puts it into the hold state (R/$\overline{C}$ going LOW) is critical, as it would be on any sample/hold amplifier. The R/$\overline{C}$ falling edge should be sharp and have minimal ringing, especially during the 20ns after it falls.

Although not normally required, it is also good practice to avoid glitching the ADS7800 while bit decisions are being made. Since the above discussion calls for a fast, clean rise and fall on R/$\overline{C}$, it makes sense to keep the rising edge of the convert pulse outside the time when bit decisions are being made. In other words, the convert pulse should either be short (under 100ns so that it transitions before the MSB decision), or relatively long (over 2.75µs to transition after the LSB decision).

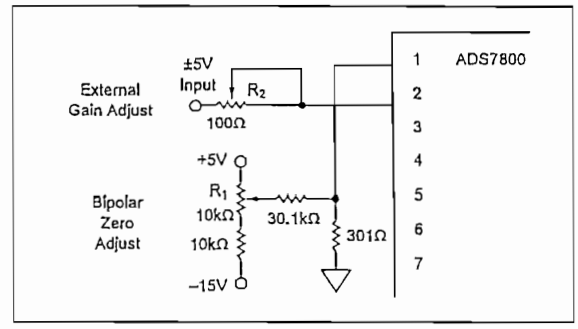


FIGURE 11. ±5V Range With External Trims.

Next, although the data outputs are forced into a Hi-Z state during conversion, fast bus transients can still be capacitively coupled into the ADS7800. If the data bus experiences fast transients during conversion, these transients can be attenuated by adding a logic buffer to the data outputs. The $\overline{BUSY}$ output can be used to enable the buffer.

Naturally, transients on the analog input signal are to be avoided, especially at times within ±20ns of R/$\overline{C}$ going LOW, when they may be trapped as part of the charge on the capacitor array. This requires careful layout of the circuit in front of the ADS7800.

Finally, in multiplexed systems, the timing on when the multiplexer is switched may affect the analog performance of the system. In most applications, the multiplexer can be switched as soon as R/$\overline{C}$ goes LOW (with appropriate delays), but this may affect the conversion if the switched signal shows glitches or significant ringing at the ADS7800 input. Whenever possible, it is safer to wait until the conversion is completed before switching the multiplexer. The extremely fast acquisition time and conversion time of the ADS7800 make this practical in many applications.

INPUT VOLTAGE RANGE AND LSB VALUES			
Input Voltage Range Defined As:		±10V	±5V
Analog Input Connected to Pin		1	2
Pin Connected to GND		2	1
One Least Significant Bit (LSB)	FSR/2^{12}	20V/2^{12}	10V/2^{12}
		4.88mV	2.44mV
OUTPUT TRANSITION VALUES			
FFE_H to FFF_H	+Full Scale	+10V–3/2LSB	+5V–3/2LSB
		+9.9927V	+4.9963V
$7FF_H$ to 800_H	Mid Scale	0V–1/2LSB	0V–1/2LSB
	(Bipolar Zero)	–2.44mV	–1.22mV
000_H to 001_H	–Full Scale	–10V+1/2LSB	–5V+1/2LSB
		–9.9976V	–4.9988V

TABLE IV. Input Voltages, Transition Values, and LSB Values.

HARRIS Semiconductor

82C54

March 1997

CMOS Programmable Interval Timer

Features

- 8MHz to 12MHz Clock Input Frequency
- Compatible with NMOS 8254
  - Enhanced Version of NMOS 8253
- Three Independent 16-Bit Counters
- Six Programmable Counter Modes
- Status Read Back Command
- Binary or BCD Counting
- Fully TTL Compatible
- Single 5V Power Supply
- Low Power
  - ICCSB 10µA
  - ICCOP .....10mA at 8MHz
- Operating Temperature Ranges
  - C82C54 .....0°C to +70°C
  - I82C54 .....-40°C to +85°C
  - M82C54 .....-55°C to +125°C

Description

The Harris 82C54 is a high performance CMOS Programmable Interval Timer manufactured using an advanced 2 micron CMOS process.

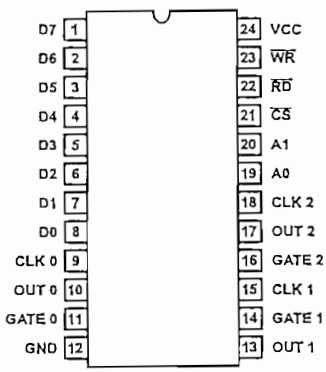
The 82C54 has three independently programmable and functional 16-bit counters, each capable of handling clock input frequencies of up to 8MHz (82C54) or 10MHz (82C54-10) or 12MHz (82C54-12).

The high speed and industry standard configuration of the 82C54 make it compatible with the Harris 80C86, 80C88, and 80C286 CMOS microprocessors along with many other industry standard processors. Six programmable timer modes allow the 82C54 to be used as an event counter, elapsed time indicator, programmable one-shot, and many other applications. Static CMOS circuit design insures low power operation.

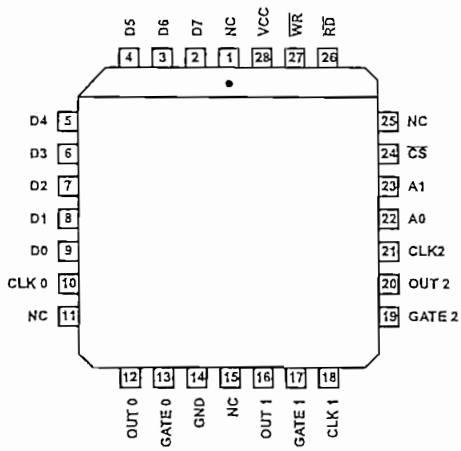
The Harris advanced CMOS process results in a significant reduction in power with performance equal to or greater than existing equivalent products.

Pinouts

82C54 (PDIP, CERDIP, SOIC)
TOP VIEW



82C54 (PLCC/CLCC)
TOP VIEW



CAUTION: These devices are sensitive to electrostatic discharge. Users should follow proper IC Handling Procedures.

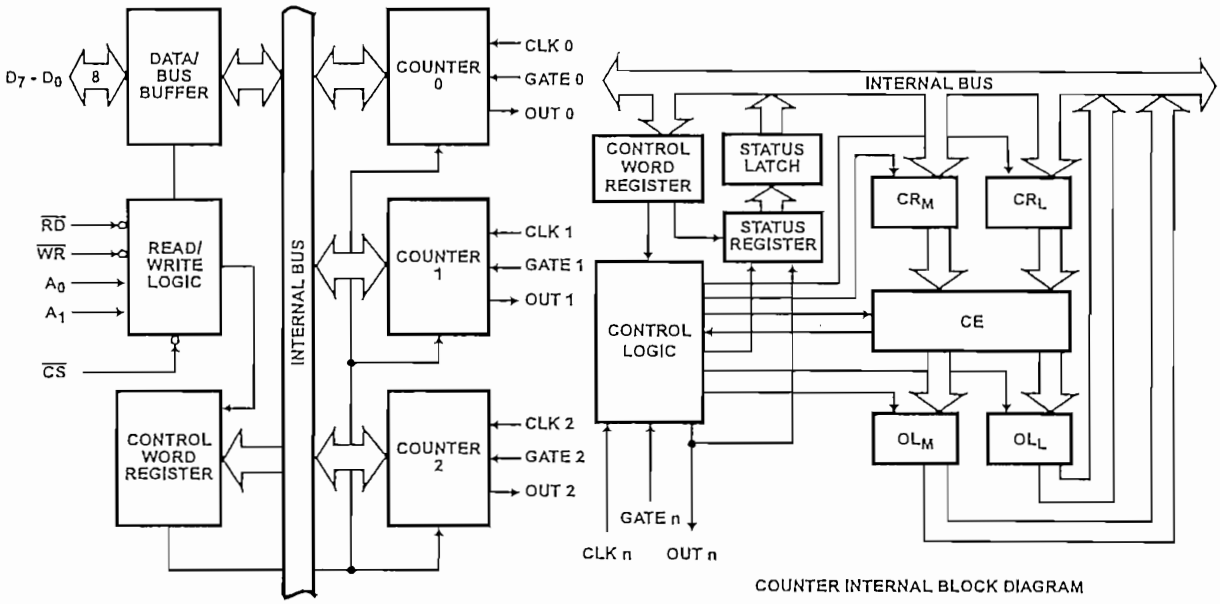
Copyright © Harris Corporation 1997

File Number 2970.1

Ordering Information

PART NUMBERS			TEMPERATURE RANGE	PACKAGE	PKG. NO.
8MHz	10MHz	12MHz			
CP82C54	CP82C54-10	CP82C54-12	0°C to +70°C	24 Lead PDIP	E24.6
IP82C54	IP82C54-10	IP82C54-12	-40°C to +85°C	24 Lead PDIP	E24.6
CS82C54	CS82C54-10	CS82C54-12	0°C to +70°C	28 Lead PLCC	N28.45
IS82C54	IS82C54-10	IS82C54-12	-40°C to +85°C	28 Lead PLCC	N28.45
CD82C54	CD82C54-10	CD82C54-12	0°C to +70°C	24 Lead CERDIP	F24.6
ID82C54	ID82C54-10	ID82C54-12	-40°C to +85°C	24 Lead CERDIP	F24.6
MD82C54/B	MD82C54-10/B	MD82C54-12/B	-55°C to +125°C	24 Lead CERDIP	F24.6
MR82C54/B	MR82C54-10/B	MR82C54-12/B	-55°C to +125°C	28 Lead CLCC	J28.A
SMD # 8406501JA	-	8406502JA	-55°C to +125°C	24 Lead CERDIP	F24.6
SMD# 84065013A	-	84065023A	-55°C to +125°C	28 Lead CLCC	J28.A
CM82C54	CM82C54-10	CM82C54-12	0°C to +70°C	24 Lead SOIC	M24.3

Functional Diagram



COUNTER INTERNAL BLOCK DIAGRAM

Pin Description

SYMBOL	DIP PIN NUMBER	TYPE	DEFINITION
D7 - D0	1 - 8	I/O	DATA: Bi-directional three-state data bus lines, connected to system data bus.
CLK 0	9	I	CLOCK 0: Clock input of Counter 0.
OUT 0	10	O	OUT 0: Output of Counter 0.
GATE 0	11	I	GATE 0: Gate input of Counter 0.
GND	12		GROUND: Power supply connection.
OUT 1	13	O	OUT 1: Output of Counter 1.
GATE 1	14	I	GATE 1: Gate input of Counter 1.
CLK 1	15	I	CLOCK 1: Clock input of Counter 1.
GATE 2	16	I	GATE 2: Gate input of Counter 2.
OUT 2	17	O	OUT 2: Output of Counter 2.

Pin Description (Continued)

SYMBOL	DIP PIN NUMBER	TYPE	DEFINITION
CLK 2	18	I	CLOCK 2: Clock input of Counter 2.
A0, A1	19 - 20	I	ADDRESS: Select inputs for one of the three counters or Control Word Register for read/write operations. Normally connected to the system address bus.
$\overline{CS}$	21	I	CHIP SELECT: A low on this input enables the 82C54 to respond to $\overline{RD}$ and $\overline{WR}$ signals. $\overline{RD}$ and $\overline{WR}$ are ignored otherwise.
$\overline{RD}$	22	I	READ: This input is low during CPU read operations.
$\overline{WR}$	23	I	WRITE: This input is low during CPU write operations.
V_{CC}	24		V_{CC}: The +5V power supply pin. A 0.1µF capacitor between pins VCC and GND is recommended for decoupling.

A1	A0	SELECTS
0	0	Counter 0
0	1	Counter 1
1	0	Counter 2
1	1	Control Word Register

Functional Description

General

The 82C54 is a programmable interval timer/counter designed for use with microcomputer systems. It is a general purpose, multi-timing element that can be treated as an array of I/O ports in the system software.

The 82C54 solves one of the most common problems in any microcomputer system, the generation of accurate time delays under software control. Instead of setting up timing loops in software, the programmer configures the 82C54 to match his requirements and programs one of the counters for the desired delay. After the desired delay, the 82C54 will interrupt the CPU. Software overhead is minimal and variable length delays can easily be accommodated.

Some of the other computer/timer functions common to microcomputers which can be implemented with the 82C54 are:

- Real time clock
- Event counter
- Digital one-shot
- Programmable rate generator
- Square wave generator
- Binary rate multiplier
- Complex waveform generator
- Complex motor controller

Data Bus Buffer

This three-state, bi-directional, 8-bit buffer is used to interface the 82C54 to the system bus (see Figure 1).

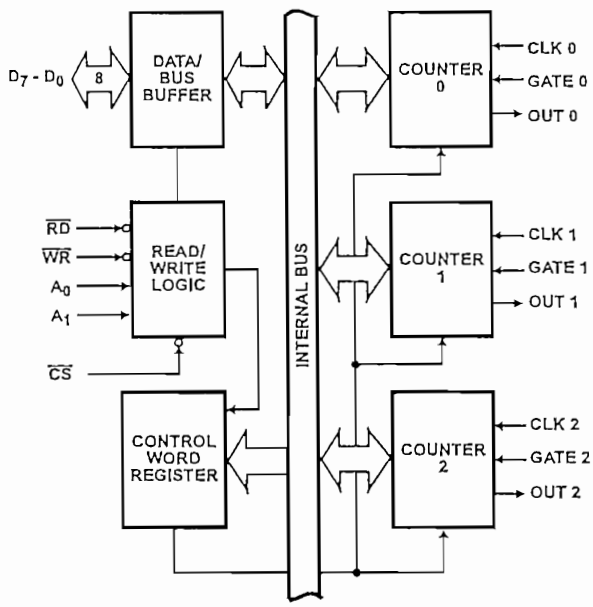


FIGURE 1. DATA BUS BUFFER AND READ/WRITE LOGIC FUNCTIONS

Read/Write Logic

The Read/Write Logic accepts inputs from the system bus and generates control signals for the other functional blocks of the 82C54. A1 and A0 select one of the three counters or the Control Word Register to be read from/written into. A "low" on the $\overline{RD}$ input tells the 82C54 that the CPU is reading one of the counters. A "low" on the $\overline{WR}$ input tells the 82C54 that the CPU is writing either a Control Word or an initial count. Both $\overline{RD}$ and $\overline{WR}$ are qualified by $\overline{CS}$; $\overline{RD}$ and $\overline{WR}$ are ignored unless the 82C54 has been selected by holding $\overline{CS}$ low.

Control Word Register

The Control Word Register (Figure 2) is selected by the Read/Write Logic when A1, A0 = 11. If the CPU then does a write operation to the 82C54, the data is stored in the Control Word Register and is interpreted as a Control Word used to define the Counter operation.

The Control Word Register can only be written to; status information is available with the Read-Back Command.

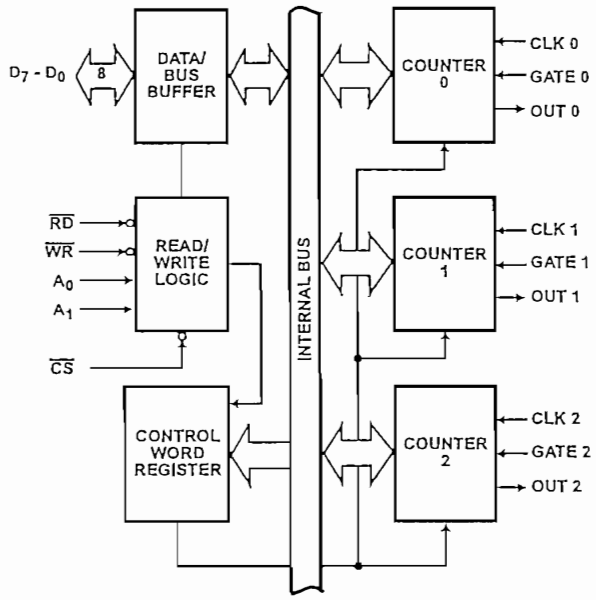


FIGURE 2. CONTROL WORD REGISTER AND COUNTER FUNCTIONS

Counter 0, Counter 1, Counter 2

These three functional blocks are identical in operation, so only a single Counter will be described. The internal block diagram of a signal counter is shown in Figure 3. The counters are fully independent. Each Counter may operate in a different Mode.

The Control Word Register is shown in the figure; it is not part of the Counter itself, but its contents determine how the Counter operates.

The status register, shown in the figure, when latched, contains the current contents of the Control Word Register and status of the output and null count flag. (See detailed explanation of the Read-Back command.)

The actual counter is labeled CE (for Counting Element). It is a 16-bit presettable synchronous down counter.

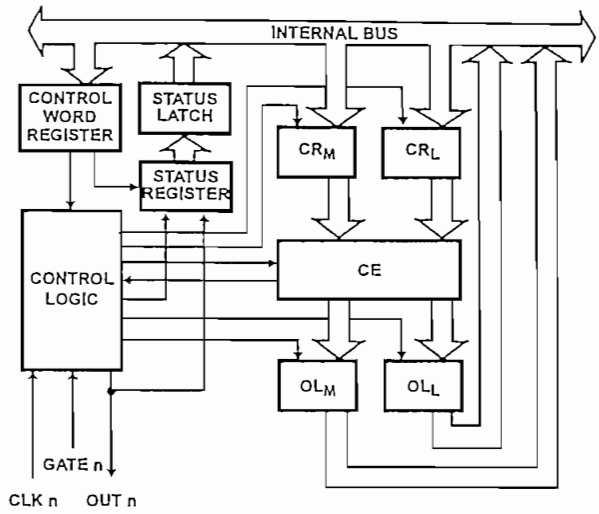


FIGURE 3. COUNTER INTERNAL BLOCK DIAGRAM

OLM and OLL are two 8-bit latches. OL stands for "Output Latch"; the subscripts M and L for "Most significant byte" and "Least significant byte", respectively. Both are normally referred to as one unit and called just OL. These latches normally "follow" the CE, but if a suitable Counter Latch Command is sent to the 82C54, the latches "latch" the present count until read by the CPU and then return to "following" the CE. One latch at a time is enabled by the counter's Control Logic to drive the internal bus. This is how the 16-bit Counter communicates over the 8-bit internal bus. Note that the CE itself cannot be read; whenever you read the count, it is the OL that is being read.

Similarly, there are two 8-bit registers called CRM and CRL (for "Count Register"). Both are normally referred to as one unit and called just CR. When a new count is written to the Counter, the count is stored in the CR and later transferred to the CE. The Control Logic allows one register at a time to be loaded from the internal bus. Both bytes are transferred to the CE simultaneously. CRM and CRL are cleared when the Counter is programmed for one byte counts (either most significant byte only or least significant byte only) the other byte will be zero. Note that the CE cannot be written into; whenever a count is written, it is written into the CR.

The Control Logic is also shown in the diagram. CLK n, GATE n, and OUT n are all connected to the outside world through the Control Logic.

82C54 System Interface

The 82C54 is treated by the system software as an array of peripheral I/O ports; three are counters and the fourth is a control register for MODE programming.

Basically, the select inputs A0, A1 connect to the A0, A1 address bus signals of the CPU. The $\overline{CS}$ can be derived directly from the address bus using a linear select method or it can be connected to the output of a decoder.

Operational Description

General

After power-up, the state of the 82C54 is undefined. The Mode, count value, and output of all Counters are undefined.

How each Counter operates is determined when it is programmed. Each Counter must be programmed before it can be used. Unused counters need not be programmed.

Programming the 82C54

Counters are programmed by writing a Control Word and then an initial count.

All Control Words are written into the Control Word Register, which is selected when A1, A0 = 11. The Control Word specifies which Counter is being programmed.

By contrast, initial counts are written into the Counters, not the Control Word Register. The A1, A0 inputs are used to select the Counter to be written into. The format of the initial count is determined by the Control Word used.

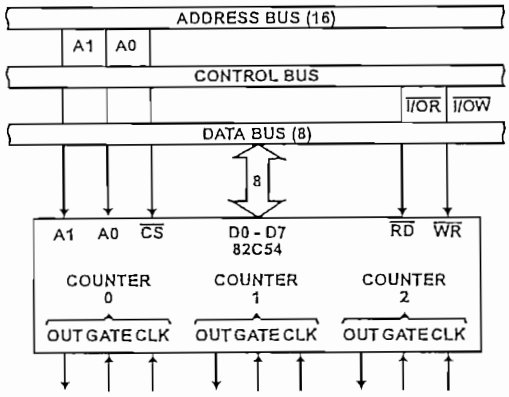


FIGURE 4. 82C54 SYSTEM INTERFACE

Write Operations

The programming procedure for the 82C54 is very flexible. Only two conventions need to be remembered:

1. For Each Counter, the Control Word must be written before the initial count is written.
2. The initial count must follow the count format specified in the Control Word (least significant byte only, most significant byte only, or least significant byte and then most significant byte).

Since the Control Word Register and the three Counters have separate addresses (selected by the A1, A0 inputs), and each Control Word specifies the Counter it applies to (SC0, SC1 bits), no special instruction sequence is required. Any programming sequence that follows the conventions above is acceptable.

Control Word Format

A1, A0 = 11; $\overline{CS}$ = 0; $\overline{RD}$ = 1; $\overline{WR}$ = 0

D7	D6	D5	D4	D3	D2	D1	D0
SC1	SC0	RW1	RW0	M2	M1	M0	BCD

SC - Select Counter

SC1	SC0	
0	0	Select Counter 0
0	1	Select Counter 1
1	0	Select Counter 2
1	1	Read-Back Command (See Read Operations)

RW - Read/Write

RW1	RW0	
0	0	Counter Latch Command (See Read Operations)
0	1	Read/Write least significant byte only.
1	0	Read/Write most significant byte only.
1	1	Read/Write least significant byte first, then most significant byte.

M - Mode

M2	M1	M0	
0	0	0	Mode 0
0	0	1	Mode 1
X	1	0	Mode 2
X	1	1	Mode 3
1	0	0	Mode 4
1	0	1	Mode 5

BCD - Binary Coded Decimal

0	Binary Counter 16-bit
1	Binary Coded Decimal (BCD) Counter (4 Decades)

NOTE: Don't Care bits (X) should be 0 to insure compatibility with future products.

Possible Programming Sequence

	A1	A0
Control Word - Counter 0	1	1
LSB of Count - Counter 0	0	0
MSB of Count - Counter 0	0	0
Control Word - Counter 1	1	1
LSB of Count - Counter 1	0	1
MSB of Count - Counter 1	0	1
Control Word - Counter 2	1	1
LSB of Count - Counter 2	1	0
MSB of Count - Counter 2	1	0

Possible Programming Sequence

	A1	A0
Control Word - Counter 0	1	1
Control Word - Counter 1	1	1
Control Word - Counter 2	1	1
LSB of Count - Counter 2	1	0

Possible Programming Sequence (Continued)

	A1	A0
LSB of Count - Counter 1	0	1
LSB of Count - Counter 0	0	0
MSB of Count - Counter 0	0	0
MSB of Count - Counter 1	0	1
MSB of Count - Counter 2	1	0

Possible Programming Sequence

	A1	A0
Control Word - Counter 2	1	1
Control Word - Counter 1	1	1
Control Word - Counter 0	1	1
LSB of Count - Counter 2	1	0
MSB of Count - Counter 2	1	0
LSB of Count - Counter 1	0	1
MSB of Count - Counter 1	0	1
LSB of Count - Counter 0	0	0
MSB of Count - Counter 0	0	0

Possible Programming Sequence

	A1	A0
Control Word - Counter 1	1	1
Control Word - Counter 0	1	1
LSB of Count - Counter 1	0	1
Control Word - Counter 2	1	1
LSB of Count - Counter 0	0	0
MSB of Count - Counter 1	0	1
LSB of Count - Counter 2	1	0
MSB of Count - Counter 0	0	0
MSB of Count - Counter 2	1	0

NOTE: In all four examples, all counters are programmed to Read/Write two-byte counts. These are only four of many programming sequences.

A new initial count may be written to a Counter at any time without affecting the Counter's programmed Mode in any way. Counting will be affected as described in the Mode definitions. The new count must follow the programmed count format.

If a Counter is programmed to read/write two-byte counts, the following precaution applies. A program must not transfer control between writing the first and second byte to another routine which also writes into that same Counter. Otherwise, the Counter will be loaded with an incorrect count.

Read Operations

It is often desirable to read the value of a Counter without disturbing the count in progress. This is easily done in the 82C54.

There are three possible methods for reading the Counters. The first is through the Read-Back command, which is explained later. The second is a simple read operation of the Counter, which is selected with the A1, A0 inputs. The only requirement is that the CLK input of the selected Counter must be inhibited by using either the GATE input or external logic. Otherwise, the count may be in process of changing when it is read, giving an undefined result.

Counter Latch Command

The other method for reading the Counters involves a special software command called the "Counter Latch Command". Like a Control Word, this command is written to the Control Word Register, which is selected when A1, A0 = 11. Also, like a Control Word, the SC0, SC1 bits select one of the three Counters, but two other bits, D5 and D4, distinguish this command from a Control Word.

A1, A0 = 11; $\overline{CS}$ = 0; $\overline{RD}$ = 1; $\overline{WR}$ = 0

D7	D6	D5	D4	D3	D2	D1	D0
SC1	SC0	0	0	X	X	X	X

SC1, SC0 - specify counter to be latched

SC1	SC0	COUNTER
0	0	0
0	1	1
1	0	2
1	1	Read-Back Command

D5, D4 - 00 designates Counter Latch Command, X - Don't Care.
NOTE: Don't Care bits (X) should be 0 to insure compatibility with future products.

The selected Counter's output latch (OL) latches the count when the Counter Latch Command is received. This count is held in the latch until it is read by the CPU (or until the Counter is reprogrammed). The count is then unlatched automatically and the OL returns to "following" the counting element (CE). This allows reading the contents of the Counters "on the fly" without affecting counting in progress. Multiple Counter Latch Commands may be used to latch more than one Counter. Each latched Counter's OL holds its count until read. Counter Latch Commands do not affect the programmed Mode of the Counter in any way.

If a Counter is latched and then, some time later, latched again before the count is read, the second Counter Latch Command is ignored. The count read will be the count at the time the first Counter Latch Command was issued.

With either method, the count must be read according to the programmed format; specifically, if the Counter is programmed for two byte counts, two bytes must be read. The two bytes do not have to be read one right after the other; read or write or programming operations of other Counters may be inserted between them.

Another feature of the 82C54 is that reads and writes of the same Counter may be interleaved; for example, if the Counter is programmed for two byte counts, the following sequence is valid.

1. Read least significant byte.
2. Write new least significant byte.
3. Read most significant byte.
4. Write new most significant byte.

If a counter is programmed to read or write two-byte counts, the following precaution applies: A program MUST NOT transfer control between reading the first and second byte to another routine which also reads from that same Counter. Otherwise, an incorrect count will be read.

Read-Back Command

The read-back command allows the user to check the count value, programmed Mode, and current state of the OUT pin and Null Count flag of the selected counter(s).

The command is written into the Control Word Register and has the format shown in Figure 5. The command applies to the counters selected by setting their corresponding bits D3, D2, D1 = 1.

A0, A1 = 11; $\overline{CS}$ = 0; $\overline{RD}$ = 1; $\overline{WR}$ = 0

D7	D6	D5	D4	D3	D2	D1	D0
1	1	$\overline{COUNT}$	$\overline{STATUS}$	CNT 2	CNT 1	CNT 0	0

D5: 0 = Latch count of selected Counter (s)
D4: 0 = Latch status of selected Counter(s)
D3: 1 = Select Counter 2
D2: 1 = Select Counter 1
D1: 1 = Select Counter 0
D0: Reserved for future expansion; Must be 0

FIGURE 5. READ-BACK COMMAND FORMAT

The read-back command may be used to latch multiple counter output latches (OL) by setting the COUNT bit D5 = 0 and selecting the desired counter(s). This signal command is functionally equivalent to several counter latch commands, one for each counter latched. Each counter's latched count is held until it is read (or the counter is reprogrammed). That counter is automatically unlatched when read, but other counters remain latched until they are read. If multiple count read-back commands are issued to the same counter without reading the count, all but the first are ignored; i.e., the count which will be read is the count at the time the first read-back command was issued.

The read-back command may also be used to latch status information of selected counter(s) by setting STATUS bit D4 = 0. Status must be latched to be read; status of a counter is accessed by a read from that counter.

The counter status format is shown in Figure 6. Bits D5 through D0 contain the counter's programmed Mode exactly as written in the last Mode Control Word. OUTPUT bit D7 contains the current state of the OUT pin. This allows the user to monitor the counter's output via software, possibly eliminating some hardware from a system.

D7	D6	D5	D4	D3	D2	D1	D0
OUTPUT	NULL COUNT	RW1	RW0	M2	M1	M0	BCD

D7: 1 = Out pin is 1
0 = Out pin is 0
D6: 1 = Null count
0 = Count available for reading
D5 - D0 = Counter programmed mode (See Control Word Formats)

FIGURE 6. STATUS BYTE

NULL COUNT bit D6 indicates when the last count written to the counter register (CR) has been loaded into the counting element (CE). The exact time this happens depends on the Mode of the counter and is described in the Mode Definitions, but until the counter is loaded into the counting element (CE), it can't be read from the counter. If the count is latched or read before this time, the count value will not reflect the new count just written. The operation of Null Count is shown below.

THIS ACTION:	CAUSES:
A. Write to the control word register:(1)	Null Count = 1
B. Write to the count register (CR):(2)	Null Count = 1
C. New count is loaded into CE (CR - CE)	Null Count = 0

(1) Only the counter specified by the control word will have its null count set to 1. Null count bits of other counters are unaffected.

(2) If the counter is programmed for two-byte counts (least significant byte then most significant byte) null count goes to 1 when the second byte is written.

If multiple status latch operations of the counter(s) are performed without reading the status, all but the first are ignored; i.e., the status that will be read is the status of the counter at the time the first status read-back command was issued.

COMMANDS								DESCRIPTION	RESULT
D7	D6	D5	D4	D3	D2	D1	D0		
1	1	0	0	0	0	1	0	Read-Back Count and Status of Counter 0	Count and Status Latched for Counter 0
1	1	1	0	0	1	0	0	Read-Back Status of Counter 1	Status Latched for Counter 1
1	1	1	0	1	1	0	0	Read-Back Status of Counters 2, 1	Status Latched for Counter 2, But Not Counter 1
1	1	0	1	1	0	0	0	Read-Back Count of Counter 2	Count Latched for Counter 2
1	1	0	0	0	1	0	0	Read-Back Count and Status of Counter 1	Count Latched for Counter 1, But Not Status
1	1	1	0	0	0	1	0	Read-Back Status of Counter 1	Command Ignored, Status Already Latched for Counter 1

FIGURE 7. READ-BACK COMMAND EXAMPLE

Both count and status of the selected counter(s) may be latched simultaneously by setting both COUNT and STATUS bits D5, D4 = 0. This is functionally the same as issuing two separate read-back commands at once, and the above discussions apply here also. Specifically, if multiple count and/or status read-back commands are issued to the same counter(s) without any intervening reads, all but the first are ignored. This is illustrated in Figure 7.

If both count and status of a counter are latched, the first read operation of that counter will return latched status, regardless of which was latched first. The next one or two reads (depending on whether the counter is programmed for one or two type counts) return latched count. Subsequent reads return unlatched count.

$\overline{CS}$	$\overline{RD}$	$\overline{WR}$	A1	A0	
0	1	0	0	0	Write into Counter 0
0	1	0	0	1	Write into Counter 1
0	1	0	1	0	Write into Counter 2
0	1	0	1	1	Write Control Word
0	0	1	0	0	Read from Counter 0
0	0	1	0	1	Read from Counter 1
0	0	1	1	0	Read from Counter 2
0	0	1	1	1	No-Operation (Three-State)
1	X	X	X	X	No-Operation (Three-State)
0	1	1	X	X	No-Operation (Three-State)

FIGURE 8. READ/WRITE OPERATIONS SUMMARY

Mode Definitions

The following are defined for use in describing the operation of the 82C54.

CLK PULSE:

A *rising* edge, then a falling edge, in that order, of a Counter's CLK input.

TRIGGER:

A rising edge of a Counter's Gate input.

COUNTER LOADING:

The transfer of a count from the CR to the CE (See "Functional Description")

Mode 0: Interrupt on Terminal Count

Mode 0 is typically used for event counting. After the Control Word is written, OUT is initially low, and will remain low until the Counter reaches zero. OUT then goes high and remains high until a new count or a new Mode 0 Control Word is written to the Counter.

GATE = 1 enables counting; GATE = 0 disables counting. GATE has no effect on OUT.

After the Control Word and initial count are written to a Counter, the initial count will be loaded on the next CLK pulse. This CLK pulse does not decrement the count, so for an initial count of N, OUT does not go high until N + 1 CLK pulses after the initial count is written.

If a new count is written to the Counter it will be loaded on the next CLK pulse and counting will continue from the new count. If a two-byte count is written, the following happens:

(1) Writing the first byte disables counting. Out is set low immediately (no clock pulse required).

(2) Writing the second byte allows the new count to be loaded on the next CLK pulse.

This allows the counting sequence to be synchronized by software. Again OUT does not go high until N + 1 CLK pulses after the new count of N is written.

If an initial count is written while GATE = 0, it will still be loaded on the next CLK pulse. When GATE goes high, OUT will go high N CLK pulses later; no CLK pulse is needed to load the counter as this has already been done.

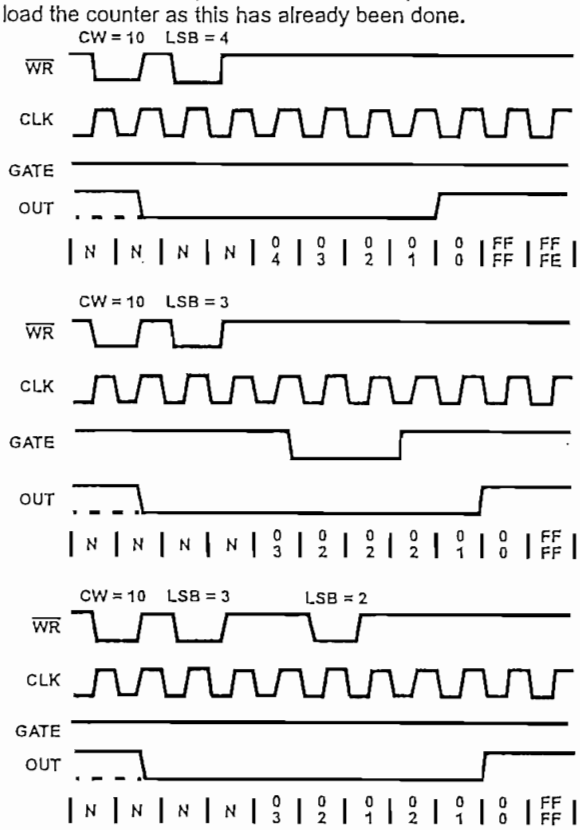


FIGURE 9. MODE 0

NOTES: The following conventions apply to all mode timing diagrams.

1. Counters are programmed for binary (not BCD) counting and for reading/writing least significant byte (LSB) only.
2. The counter is always selected ($\overline{CS}$ always low).
3. CW stands for "Control Word"; CW = 10 means a control word of 10, Hex is written to the counter.
4. LSB stands for Least significant "byte" of count.
5. Numbers below diagrams are count values. The lower number is the least significant byte. The upper number is the most significant byte. Since the counter is programmed to read/write LSB only, the most significant byte cannot be read.
6. N stands for an undefined count.
7. Vertical lines show transitions between count values.

Mode 1: Hardware Retriggerable One-Shot

OUT will be initially high. OUT will go low on the CLK pulse following a trigger to begin the one-shot pulse, and will remain low until the Counter reaches zero. OUT will then go high and remain high until the CLK pulse after the next trigger.

After writing the Control Word and initial count, the Counter is armed. A trigger results in loading the Counter and setting OUT low on the next CLK pulse, thus starting the one-shot pulse N CLK cycles in duration. The one-shot is retriggerable, hence OUT will remain low for N CLK pulses after any trigger. The one-shot pulse can be repeated without rewriting the same count into the counter. GATE has no effect on OUT.

If a new count is written to the Counter during a one-shot pulse, the current one-shot is not affected unless the Counter is retriggerable. In that case, the Counter is loaded with the new count and the one-shot pulse continues until the new count expires.

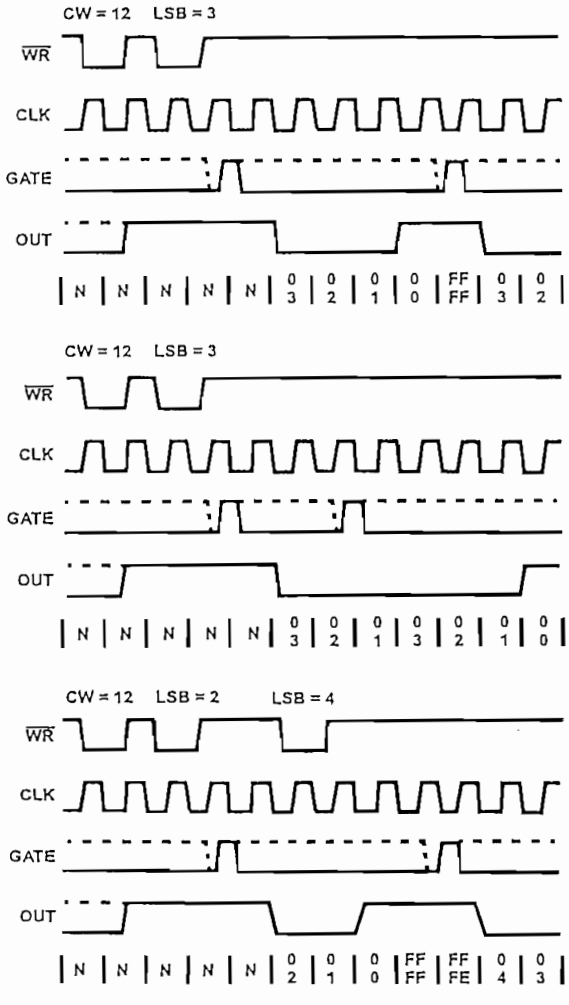


FIGURE 10. MODE 1

Mode 2: Rate Generator

This Mode functions like a divide-by-N counter. It is typically used to generate a Real Time Clock Interrupt. OUT will initially be high. When the initial count has decremented to 1, OUT goes low for one CLK pulse. OUT then goes high again, the Counter reloads the initial count and the process is repeated. Mode 2 is periodic; the same sequence is repeated indefinitely. For an initial count of N, the sequence repeats every N CLK cycles.

GATE = 1 enables counting; GATE = 0 disables counting. If GATE goes low during an output pulse, OUT is set high immediately. A trigger reloads the Counter with the initial count on the next CLK pulse; OUT goes low N CLK pulses after the trigger. Thus the GATE input can be used to synchronize the Counter.

After writing a Control Word and initial count, the Counter will be loaded on the next CLK pulse. OUT goes low N CLK pulses after the initial count is written. This allows the Counter to be synchronized by software also.

Writing a new count while counting does not affect the current counting sequence. If a trigger is received after writing a new count but before the end of the current period, the Counter will be loaded with the new count on the next CLK pulse and counting will continue from the end of the current counting cycle.

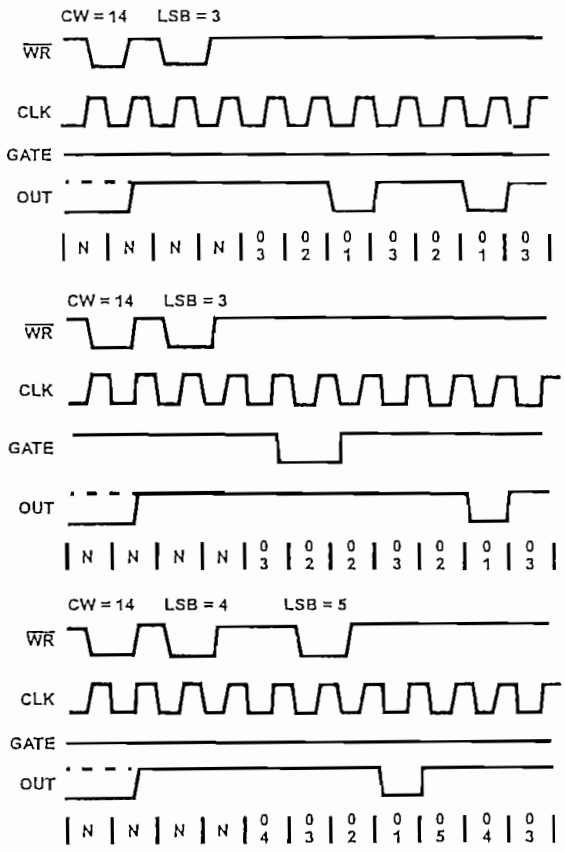


FIGURE 11. MODE 2

Mode 3: Square Wave Mode

Mode 3 is typically used for Baud rate generation. Mode 3 is similar to Mode 2 except for the duty cycle of OUT. OUT will initially be high. When half the initial count has expired, OUT goes low for the remainder of the count. Mode 3 is periodic; the sequence above is repeated indefinitely. An initial count of N results in a square wave with a period of N CLK cycles.

GATE = 1 enables counting; GATE = 0 disables counting. If GATE goes low while OUT is low, OUT is set high immediately; no CLK pulse is required. A trigger reloads the Counter with the initial count on the next CLK pulse. Thus the GATE input can be used to synchronize the Counter.

After writing a Control Word and initial count, the Counter will be loaded on the next CLK pulse. This allows the Counter to be synchronized by software also.

Writing a new count while counting does not affect the current counting sequence. If a trigger is received after writing a new count but before the end of the current half-cycle of the square wave, the Counter will be loaded with the new count on the next CLK pulse and counting will continue from the new count. Otherwise, the new count will be loaded at the end of the current half-cycle.

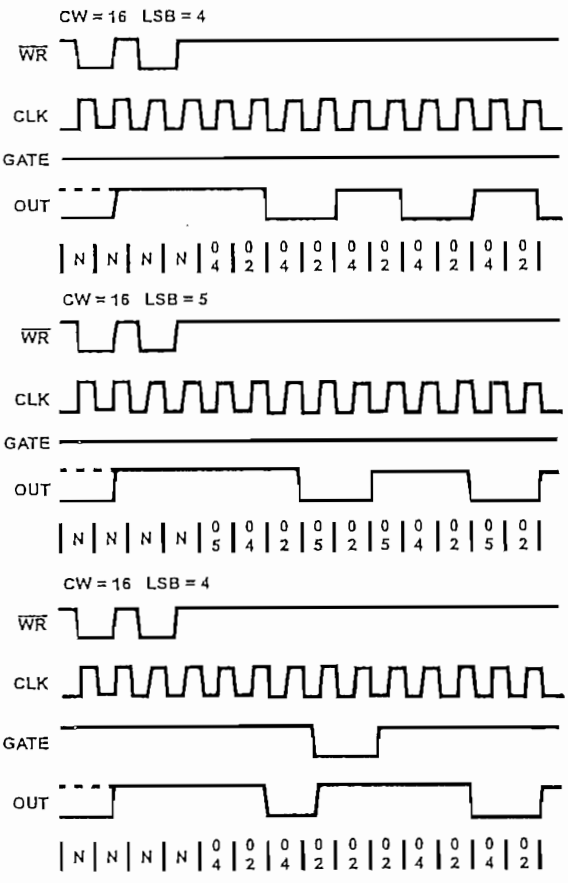


FIGURE 12. MODE 3

Mode 3 is Implemented as Follows:

EVEN COUNTS: OUT is initially high. The initial count is loaded on one CLK pulse and then is decremented by two on succeeding CLK pulses. When the count expires, OUT changes value and the Counter is reloaded with the initial count. The above process is repeated indefinitely.

ODD COUNTS: OUT is initially high. The initial count is loaded on one CLK pulse, decremented by one on the next CLK pulse, and then decremented by two on succeeding CLK pulses. When the count expires, OUT goes low and the Counter is reloaded with the initial count. The count is decremented by three on the next CLK pulse, and then by two on succeeding CLK pulses. When the count expires, OUT goes high again and the Counter is reloaded with the initial count. The above process is repeated indefinitely. So for odd counts, OUT will be high for (N + 1)/2 counts and low for (N - 1)/2 counts.

Mode 4: Software Triggered Mode

OUT will be initially high. When the initial count expires, OUT will go low for one CLK pulse then go high again. The counting sequence is "Triggered" by writing the initial count.

GATE = 1 enables counting; GATE = 0 disables counting. GATE has no effect on OUT.

After writing a Control Word and initial count, the Counter will be loaded on the next CLK pulse. This CLK pulse does not decrement the count, so for an initial count of N, OUT does not strobe low until N + 1 CLK pulses after the initial count is written.

If a new count is written during counting, it will be loaded on the next CLK pulse and counting will continue from the new count. If a two-byte count is written, the following happens:

(1) Writing the first byte has no effect on counting.

(2) Writing the second byte allows the new count to be loaded on the next CLK pulse.

This allows the sequence to be "retriggered" by software. OUT strobes low N + 1 CLK pulses after the new count of N is written.

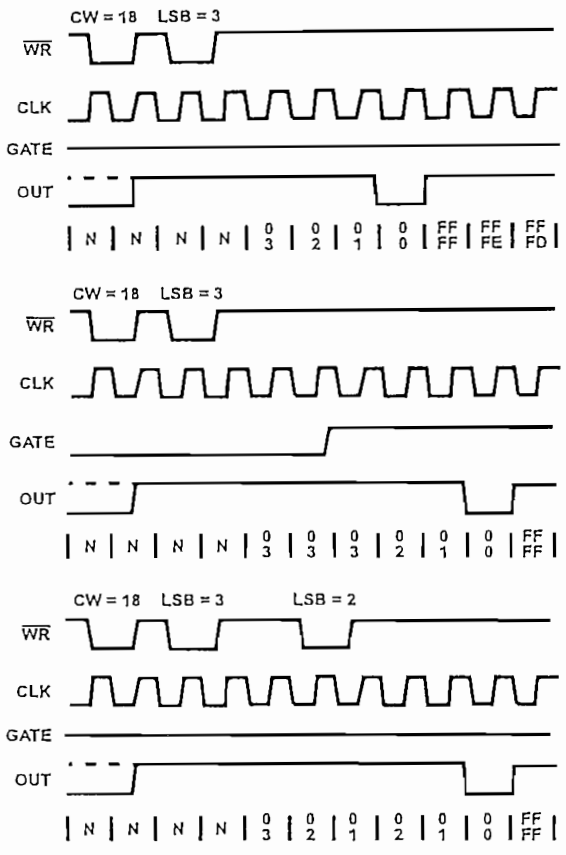


FIGURE 13. MODE 4

Mode 5: Hardware Triggered Strobe (Retriggerable)

OUT will initially be high. Counting is triggered by a rising edge of GATE. When the initial count has expired, OUT will go low for one CLK pulse and then go high again.

After writing the Control Word and initial count, the counter will not be loaded until the CLK pulse after a trigger. This CLK pulse does not decrement the count, so for an initial count of N, OUT does not strobe low until N + 1 CLK pulses after trigger.

A trigger results in the Counter being loaded with the initial count on the next CLK pulse. The counting sequence is triggerable. OUT will not strobe low for N + 1 CLK pulses after any trigger GATE has no effect on OUT.

If a new count is written during counting, the current counting sequence will not be affected. If a trigger occurs after the new count is written but before the current count expires, the Counter will be loaded with new count on the next CLK pulse and counting will continue from there.

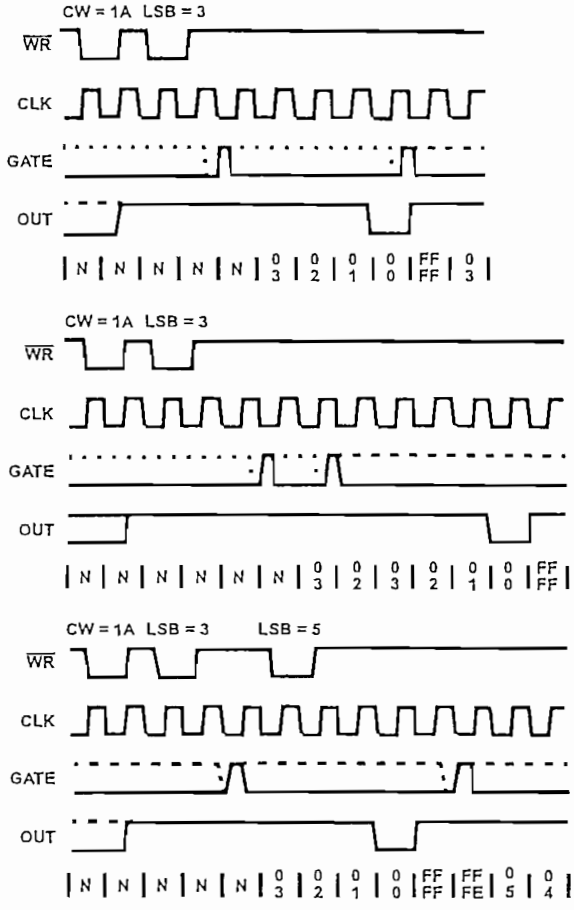


FIGURE 14. MODE 5

Operation Common to All Modes

Programming

When a Control Word is written to a Counter, all Control Logic, is immediately reset and OUT goes to a known initial state; no CLK pulses are required for this.

Gate

The GATE input is always sampled on the rising edge of CLK. In Modes 0, 2, 3 and 4 the GATE input is level sensitive, and logic level is sampled on the rising edge of CLK. In modes 1, 2, 3 and 5 the GATE input is rising-edge sensitive. In these Modes, a rising edge of Gate (trigger) sets an edge-sensitive flip-flop in the Counter. This flip-flop is then sampled on the next rising edge of CLK. The flip-flop is reset immediately after it is sampled. In this way, a trigger will be detected no matter when it occurs - a high logic level does not have to be maintained until the next rising edge of CLK. Note that in Modes 2 and 3, the GATE input is both edge- and level-sensitive.

Counter

New counts are loaded and Counters are decremented on the falling edge of CLK.

The largest possible initial count is 0; this is equivalent to 2^{16} for binary counting and 10^4 for BCD counting.

The counter does not stop when it reaches zero. In Modes 0, 1, 4, and 5 the Counter "wraps around" to the highest count, either FFFF hex for binary counting or 9999 for BCD counting, and continues counting. Modes 2 and 3 are periodic; the Counter reloads itself with the initial count and continues counting from there.

SIGNAL STATUS MODES	LOW OR GOING LOW	RISING	HIGH
0	Disables Counting	-	Enables Counting
1	-	1) Initiates Counting 2) Resets output after next clock	-
2	1) Disables counting 2) Sets output immediately high	Initiates Counting	Enables Counting
3	1) Disables counting 2) Sets output immediately high	Initiates Counting	Enables Counting
4	1) Disables Counting	-	Enables Counting
5	-	Initiates Counting	-

FIGURE 15. GATE PIN OPERATIONS SUMMARY

MODE	MIN COUNT	MAX COUNT
0	1	0
1	1	0
2	2	0
3	2	0
4	1	0
5	1	0

NOTE: 0 is equivalent to 2^{16} for binary counting and 10^4 for BCD counting.

FIGURE 16. MINIMUM AND MAXIMUM INITIAL COUNTS

Absolute Maximum Ratings

Supply Voltage ..... +8.0V
Input, Output or I/O Voltage ..... GND-0.5V to V_{CC} +0.5V
ESD Classification ..... Class 1

Operating Conditions

Operating Voltage Range ..... +4.5V to +5.5V
Operating Temperature Range
C82C54, C82C54-10, -12 ..... 0°C to +70°C
I82C54, I82C54-10, -12 ..... -40°C to +85°C
M82C54, M82C54-10, -12 ..... -55°C to +125°C

Thermal Information

Thermal Resistance (Typical)	θ_{JA} (°C/W)	θ_{JC} (°C/W)
CERDIP Package	55	12
CLCC Package	65	14
PDIP Package	60	N/A
PLCC Package	65	N/A
SOIC Package	75	N/A

Storage Temperature Range ..... -65°C to +150°C
Maximum Junction Temperature Ceramic Package ..... +175°C
Maximum Junction Temperature Plastic Package ..... +150°C
Maximum Lead Temperature Package (Soldering 10s) .... +300°C
(PLCC and SOIC - Lean Tips Only)

Die Characteristics

Gate Count ..... 2250 Gates

CAUTION: Stresses above those listed in "Absolute Maximum Ratings" may cause permanent damage to the device. This is a stress only rating and operation of the device at these or any other conditions above those indicated in the operational sections of this specification is not implied.

DC Electrical Specifications

V_{CC} = +5.0V ± 10%, T_A = 0°C to +70°C (C82C54, C82C54-10, C82C54-12)
T_A = -40°C to +85°C (I82C54, I82C54-10, I82C54-12)
T_A = -55°C to +125°C (M82C54, M82C54-10, M82C54-12

SYMBOL	PARAMETER	MIN	MAX	UNITS	TEST CONDITIONS
VIH	Logical One Input Voltage	2.0	-	V	C82C54, I82C54
		2.2	-	V	M82C54
VIL	Logical Zero Input Voltage	-	0.8	V	
VOH	Output HIGH Voltage	3.0	-	V	IOH = -2.5mA
		V_{CC} -0.4	-	V	IOH = -100µA
VOL	Output LOW Voltage	-	0.4	V	IOL = +2.5mA
II	Input Leakage Current	-1	+1	µA	VIN = GND or V_{CC} DIP Pins 9,11,14-16,18-23
IO	Output Leakage Current	-10	+10	µA	VOUT = GND or V_{CC} DIP Pins 1-8
ICCSB	Standby Power Supply Current	-	10	µA	V_{CC} = 5.5V, VIN = GND or V_{CC}, Outputs Open, Counters Programmed
ICCOP	Operating Power Supply Current	-	10	mA	V_{CC} = 5.5V, CLK0 = CLK1 = CLK2 = 8MHz, VIN = GND or V_{CC}, Outputs Open

Capacitance

T_A = +25°C; All Measurements Referenced to Device GND, Note 1

SYMBOL	PARAMETER	TYP	UNITS	TEST CONDITIONS
CIN	Input Capacitance	20	pF	FREQ = 1MHz
COUT	Output Capacitance	20	pF	FREQ = 1MHz
CI/O	I/O Capacitance	20	pF	FREQ = 1MHz

NOTE:

1. Not tested, but characterized at initial design and at major process/design changes.

AC Electrical Specifications

V_{CC} = +5.0V ± 10%, T_A = 0°C to +70°C (C82C54, C82C54-10, C82C54-12)
T_A = -40°C to +85°C (I82C54, I82C54-10, I82C54-12)
T_A = -55°C to +125°C (M82C54, M82C54-10, M82C54-12)

SYMBOL		PARAMETER	82C54		82C54-10		82C54-12		UNITS	TEST CONDITIONS
			MIN	MAX	MIN	MAX	MIN	MAX		
READ CYCLE										
(1)	TAR	Address Stable Before $\overline{RD}$	30	-	25	-	25	-	ns	1
(2)	TSR	$\overline{CS}$ Stable Before $\overline{RD}$	0	-	0	-	0	-	ns	1
(3)	TRA	Address Hold Time After $\overline{RD}$	0	-	0	-	0	-	ns	1
(4)	TRR	$\overline{RD}$ Pulse Width	150	-	95	-	95	-	ns	1
(5)	TRD	Data Delay from $\overline{RD}$	-	120	-	85	-	85	ns	1
(6)	TAD	Data Delay from Address	-	210	-	185	-	185	ns	1
(7)	TDF	$\overline{RD}$ to Data Floating	5	85	5	65	5	65	ns	2, Note 1
(8)	TRV	Command Recovery Time	200	-	165	-	165	-	ns	
WRITE CYCLE										
(9)	TAW	Address Stable Before $\overline{WR}$	0	-	0	-	0	-	ns	
(10)	TSW	$\overline{CS}$ Stable Before $\overline{WR}$	0	-	0	-	0	-	ns	
(11)	TWA	Address Hold Time After $\overline{WR}$	0	-	0	-	0	-	ns	
(12)	TWW	$\overline{WR}$ Pulse Width	95	-	95	-	95	-	ns	
(13)	TDW	Data Setup Time Before $\overline{WR}$	140	-	95	-	95	-	ns	
(14)	TWD	Data Hold Time After $\overline{WR}$	25	-	0	-	0	-	ns	
(15)	TRV	Command Recovery Time	200	-	165	-	165	-	ns	
CLOCK AND GATE										
(16)	TCLK	Clock Period	125	DC	100	DC	80	DC	ns	1
(17)	TPWH	High Pulse Width	60	-	30	-	30	-	ns	1
(18)	TPWL	Low Pulse Width	60	-	40	-	30	-	ns	1
(19)	TR	Clock Rise Time	-	25	-	25	-	25	ns	
(20)	TF	Clock Fall Time	-	25	-	25	-	25	ns	
(21)	TGW	Gate Width High	50	-	50	-	50	-	ns	1
(22)	TGL	Gate Width Low	50	-	50	-	50	-	ns	1
(23)	TGS	Gate Setup Time to CLK	50	-	40	-	40	-	ns	1
(24)	TGH	Gate Hold Time After CLK	50	-	50	-	50	-	ns	1
(25)	TOD	Output Delay from CLK	-	150	-	100	-	100	ns	1
(26)	TODG	Output Delay from Gate	-	120	-	100	-	100	ns	1
(27)	TWO	OUT Delay from Mode Write	-	260	-	240	-	240	ns	1
(28)	TWC	CLK Delay for Loading	0	55	0	55	0	55	ns	1
(29)	TWG	Gate Delay for Sampling	-5	40	-5	40	-5	40	ns	1
(30)	TCL	CLK Setup for Count Latch	-40	40	-40	40	-40	40	ns	1

NOTE:

1. Not tested, but characterized at initial design and at major process/design changes.

Timing Waveforms

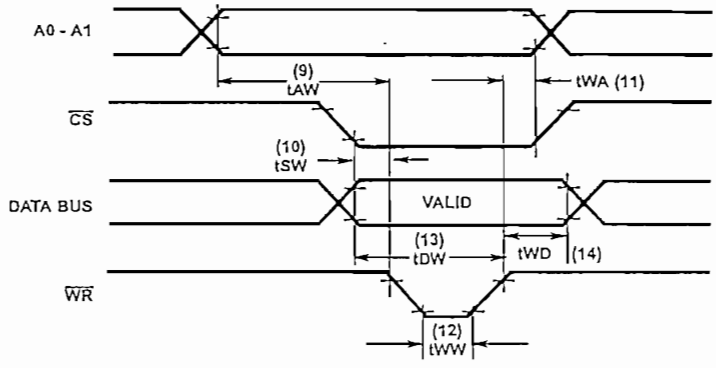


FIGURE 17. WRITE

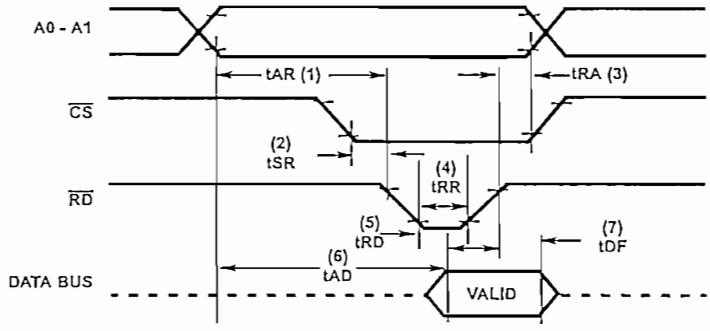


FIGURE 18. READ

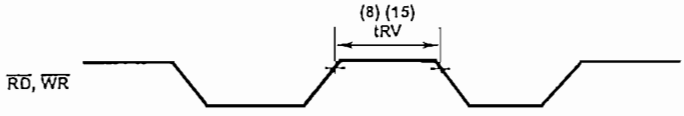


FIGURE 19. RECOVERY

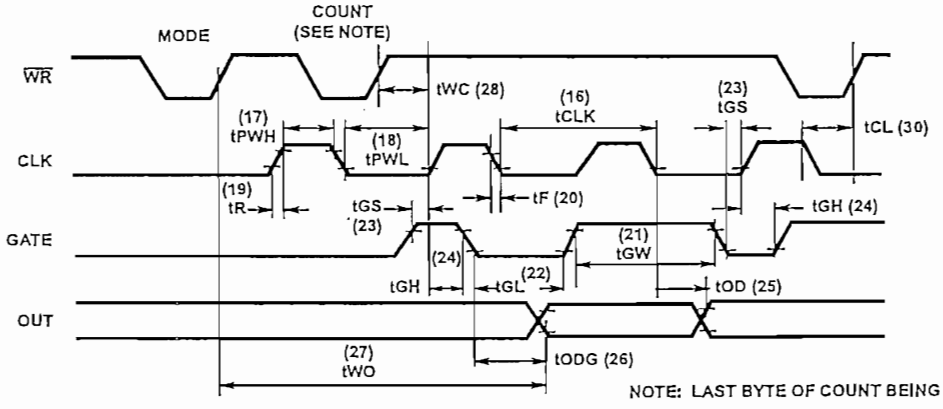
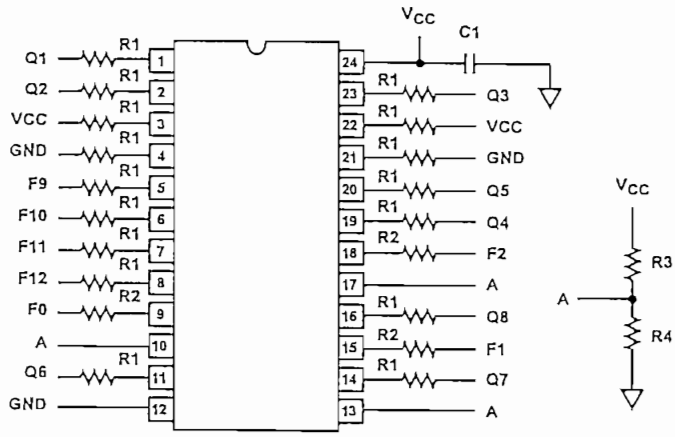


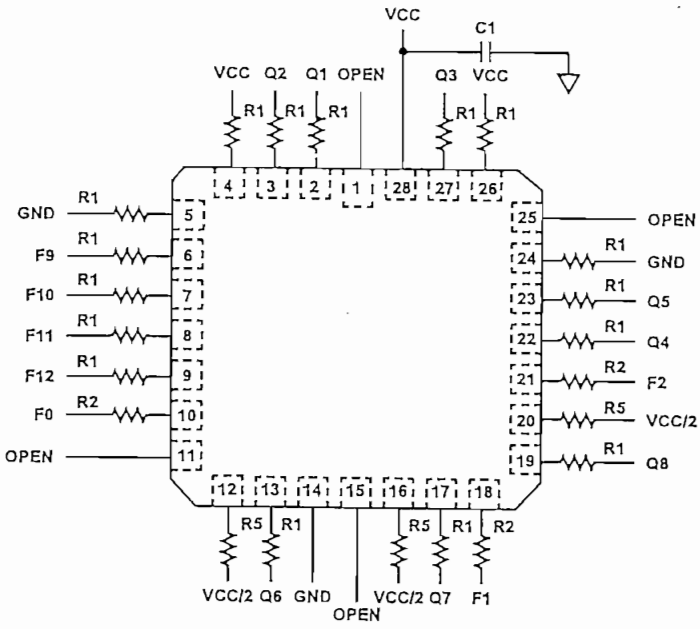
FIGURE 20. CLOCK AND GATE

Burn-In Circuits

MD 82C54 CERDIP



MR 82C54 CLCC



NOTES:

1. V_{CC} = 5.5V ± 0.5V
2. GND = 0V
3. VIH = 4.5V ±10%
4. VIL = -0.2V to 0.4V
5. R1 = 47kΩ ±5%
6. R2 = 1.0kΩ ±5%
7. R3 = 2.7kΩ ±5%
8. R4 = 1.8kΩ ±5%
9. R5 = 1.2kΩ ±5%
10. C1 = 0.01µF Min
11. F0 = 100kHz ±10%
12. F1 = F0/2, F2 = F1/2, ...F12 = F11/2

Die Characteristics

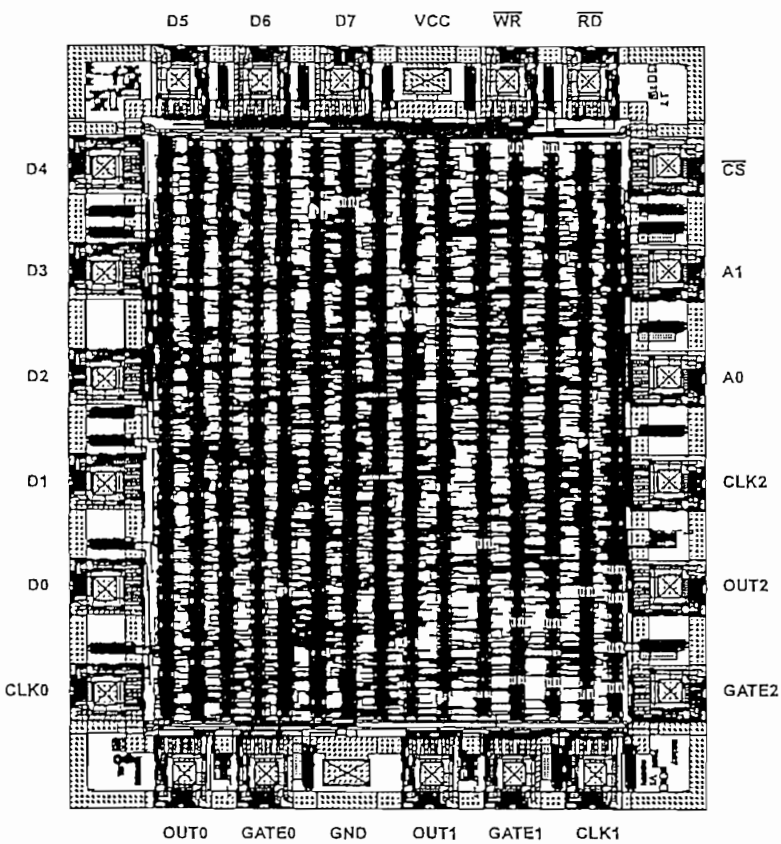
DIE DIMENSIONS:
129mils x 155mils x 19mils
(3270µm x 3940µm x 483µm)

METALLIZATION:
Type: Si-Al-Cu
Thickness: Metal 1: 8kÅ ± 0.75kÅ
Metal 2: 12kÅ ± 1.0kÅ

GLASSIVATION:
Type: Nitrox
Thickness: 10kÅ ± 3.0kÅ

Metallization Mask Layout

82C54



HARRIS Semiconductor

82C55A

CMOS Programmable Peripheral Interface

June 1998

Features

- Pin Compatible with NMOS 8255A
- 24 Programmable I/O Pins
- Fully TTL Compatible
- High Speed, No "Wait State" Operation with 5MHz and 8MHz 80C86 and 80C88
- Direct Bit Set/Reset Capability
- Enhanced Control Word Read Capability
- L7 Process
- 2.5mA Drive Capability on All I/O Ports
- Low Standby Power (ICCSB) 10µA

Ordering Information

PART NUMBERS		PACKAGE	TEMPERATURE RANGE	PKG. NO.
5MHz	8MHz			
CP82C55A-5	CP82C55A	40 Ld PDIP	0°C to 70°C	E40.6
IP82C55A-5	IP82C55A		-40°C to 85°C	E40.6
CS82C55A-5	CS82C55A	44 Ld PLCC	0°C to 70°C	N44.65
IS82C55A-5	IS82C55A		-40°C to 85°C	N44.65
CD82C55A-5	CD82C55A	40 Ld CERDIP	0°C to 70°C	F40.6
ID82C55A-5	ID82C55A		-40°C to 85°C	F40.6
MD82C55A-5/B	MD82C55A/B		-55°C to 125°C	F40.6
8406601QA	8406602QA	SMD#		F40.6
MR82C55A-5/B	MR82C55A/B	44 Pad CLCC	-55°C to 125°C	J44.A
8406601XA	8406602XA	SMD#		J44.A

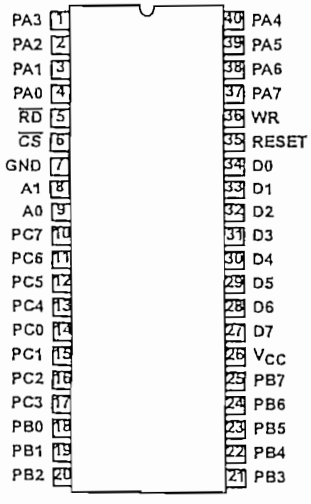
Description

The Harris 82C55A is a high performance CMOS version of the industry standard 8255A and is manufactured using a self-aligned silicon gate CMOS process (Scaled SAJI IV). It is a general purpose programmable I/O device which may be used with many different microprocessors. There are 24 I/O pins which may be individually programmed in 2 groups of 12 and used in 3 major modes of operation. The high performance and industry standard configuration of the 82C55A make it compatible with the 80C86, 80C88 and other microprocessors.

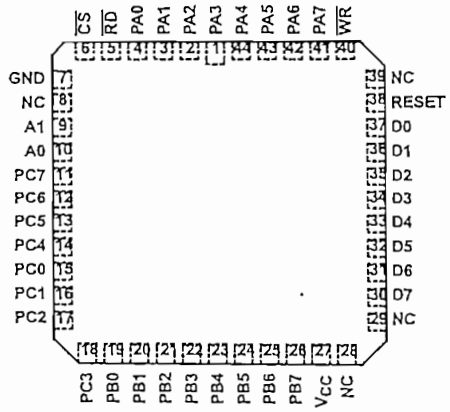
Static CMOS circuit design insures low operating power. TTL compatibility over the full military temperature range and bus hold circuitry eliminate the need for pull-up resistors. The Harris advanced SAJI process results in performance equal to or greater than existing functionally equivalent products at a fraction of the power.

Pinouts

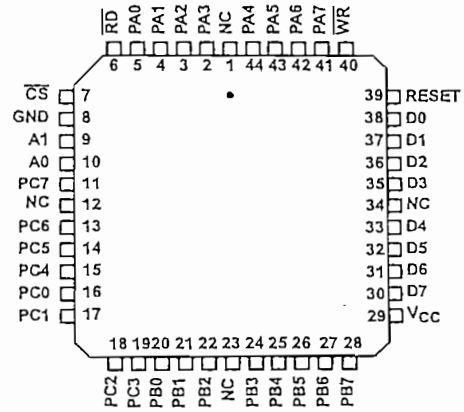
82C55A (DIP)
TOP VIEW



82C55A (CLCC)
TOP VIEW



82C55A (PLCC)
TOP VIEW



CAUTION: These devices are sensitive to electrostatic discharge. Users should follow proper IC Handling Procedures.

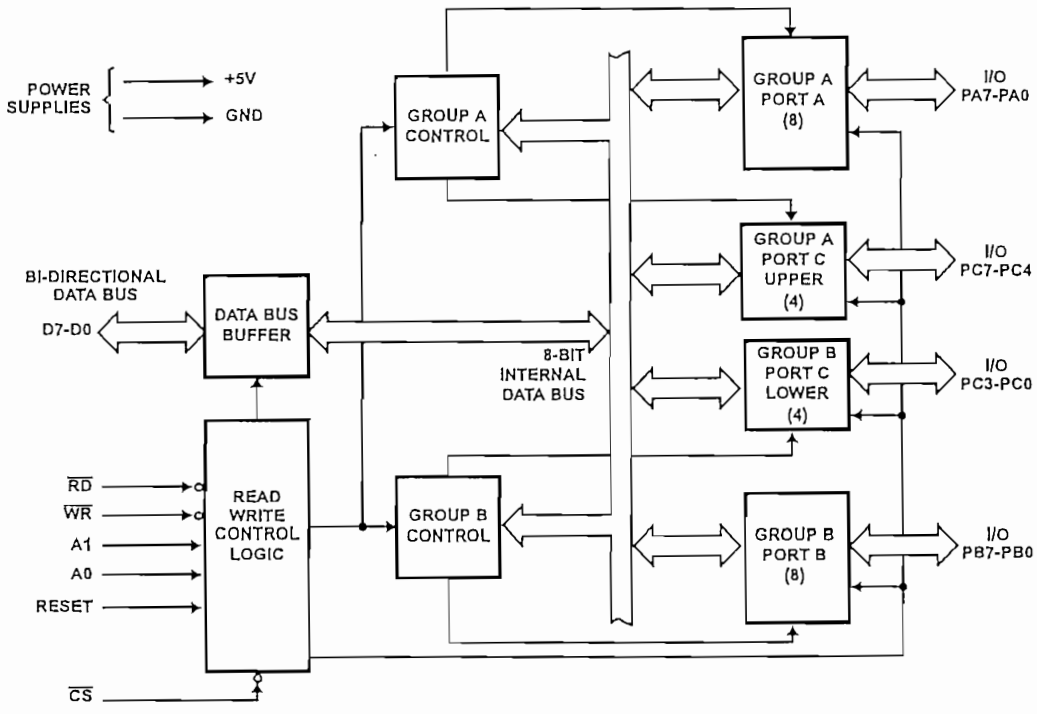
Copyright © Harris Corporation 1998

File Number 2969.2

Pin Description

SYMBOL	PIN NUMBER	TYPE	DESCRIPTION
V_{CC}	26		V_{CC}: The +5V power supply pin. A 0.1µF capacitor between pins 26 and 7 is recommended for decoupling.
GND	7		GROUND
D0-D7	27-34	I/O	DATA BUS: The Data Bus lines are bidirectional three-state pins connected to the system data bus.
RESET	35	I	RESET: A high on this input clears the control register and all ports (A, B, C) are set to the input mode with the "Bus Hold" circuitry turned on.
$\overline{CS}$	6	I	CHIP SELECT: Chip select is an active low input used to enable the 82C55A onto the Data Bus for CPU communications.
$\overline{RD}$	5	I	READ: Read is an active low input control signal used by the CPU to read status information or data via the data bus.
$\overline{WR}$	36	I	WRITE: Write is an active low input control signal used by the CPU to load control words and data into the 82C55A.
A0-A1	8, 9	I	ADDRESS: These input signals, in conjunction with the $\overline{RD}$ and $\overline{WR}$ inputs, control the selection of one of the three ports or the control word register. A0 and A1 are normally connected to the least significant bits of the Address Bus A0, A1.
PA0-PA7	1-4, 37-40	I/O	PORT A: 8-bit input and output port. Both bus hold high and bus hold low circuitry are present on this port.
PB0-PB7	18-25	I/O	PORT B: 8-bit input and output port. Bus hold high circuitry is present on this port.
PC0-PC7	10-17	I/O	PORT C: 8-bit input and output port. Bus hold circuitry is present on this port.

Functional Diagram



Functional Description

Data Bus Buffer

This three-state bi-directional 8-bit buffer is used to interface the 82C55A to the system data bus. Data is transmitted or received by the buffer upon execution of input or output instructions by the CPU. Control words and status information are also transferred through the data bus buffer.

Read/Write and Control Logic

The function of this block is to manage all of the internal and external transfers of both Data and Control or Status words. It accepts inputs from the CPU Address and Control busses and in turn, issues commands to both of the Control Groups.

(CS) Chip Select. A "low" on this input pin enables the communcation between the 82C55A and the CPU.

(RD) Read. A "low" on this input pin enables 82C55A to send the data or status information to the CPU on the data bus. In essence, it allows the CPU to "read from" the 82C55A.

(WR) Write. A "low" on this input pin enables the CPU to write data or control words into the 82C55A.

(A0 and A1) Port Select 0 and Port Select 1. These input signals, in conjunction with the RD and WR inputs, control the selection of one of the three ports or the control word register. They are normally connected to the least significant bits of the address bus (A0 and A1).

82C55A BASIC OPERATION

A1	A0	RD	WR	CS	INPUT OPERATION (READ)
0	0	0	1	0	Port A → Data Bus
0	1	0	1	0	Port B → Data Bus
1	0	0	1	0	Port C → Data Bus
1	1	0	1	0	Control Word → Data Bus
					OUTPUT OPERATION (WRITE)
0	0	1	0	0	Data Bus → Port A
0	1	1	0	0	Data Bus → Port B
1	0	1	0	0	Data Bus → Port C
1	1	1	0	0	Data Bus → Control
					DISABLE FUNCTION
X	X	X	X	1	Data Bus → Three-State
X	X	1	1	0	Data Bus → Three-State

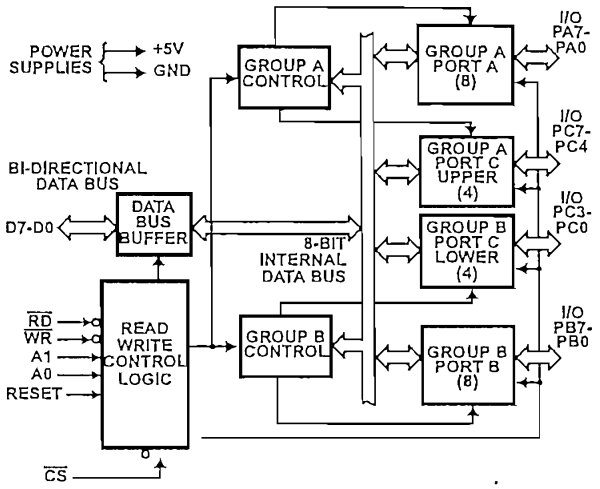


FIGURE 1. 82C55A BLOCK DIAGRAM. DATA BUS BUFFER, READ/WRITE, GROUP A & B CONTROL LOGIC FUNCTIONS

(RESET) Reset. A "high" on this input initializes the control register to 9Bh and all ports (A, B, C) are set to the input mode. "Bus hold" devices internal to the 82C55A will hold the I/O port inputs to a logic "1" state with a maximum hold current of 400μA.

Group A and Group B Controls

The functional configuration of each port is programmed by the systems software. In essence, the CPU "outputs" a control word to the 82C55A. The control word contains information such as "mode", "bit set", "bit reset", etc., that initializes the functional configuration of the 82C55A.

Each of the Control blocks (Group A and Group B) accepts "commands" from the Read/Write Control logic, receives "control words" from the internal data bus and issues the proper commands to its associated ports.

Control Group A - Port A and Port C upper (C7 - C4)

Control Group B - Port B and Port C lower (C3 - C0)

The control word register can be both written and read as shown in the "Basic Operation" table. Figure 4 shows the control word format for both Read and Write operations. When the control word is read, bit D7 will always be a logic "1", as this implies control word mode information.

Ports A, B, and C

The 82C55A contains three 8-bit ports (A, B, and C). All can be configured to a wide variety of functional characteristics by the system software but each has its own special features or "personality" to further enhance the power and flexibility of the 82C55A.

Port A One 8-bit data output latch/buffer and one 8-bit data input latch. Both "pull-up" and "pull-down" bus-hold devices are present on Port A. See Figure 2A.

Port B One 8-bit data input/output latch/buffer and one 8-bit data input buffer. See Figure 2B.

Port C One 8-bit data output latch/buffer and one 8-bit data input buffer (no latch for input). This port can be divided into two 4-bit ports under the mode control. Each 4-bit port contains a 4-bit latch and it can be used for the control signal output and status signal inputs in conjunction with ports A and B. See Figure 2B.

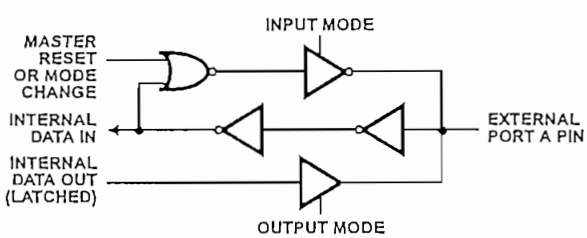


FIGURE 2A. PORT A BUS-HOLD CONFIGURATION

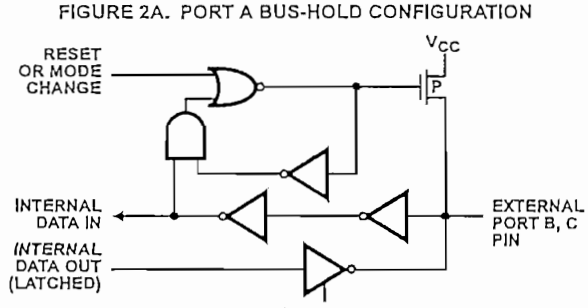


FIGURE 2B. PORT B AND C BUS-HOLD CONFIGURATION

FIGURE 2. BUS-HOLD CONFIGURATION

Operational Description

Mode Selection

There are three basic modes of operation than can be selected by the system software:

Mode 0 - Basic Input/Output
Mode 1 - Strobed Input/Output
Mode 2 - Bi-directional Bus

When the reset input goes "high", all ports will be set to the input mode with all 24 port lines held at a logic "one" level by internal bus hold devices. After the reset is removed, the 82C55A can remain in the input mode with no additional initialization required. This eliminates the need to pullup or pulldown resistors in all-CMOS designs. The control word register will contain 9Bh. During the execution of the system program, any of the other modes may be selected using a single output instruction. This allows a single 82C55A to service a variety of peripheral devices with a simple software maintenance routine. Any port programmed as an output port is initialized to all zeros when the control word is written.

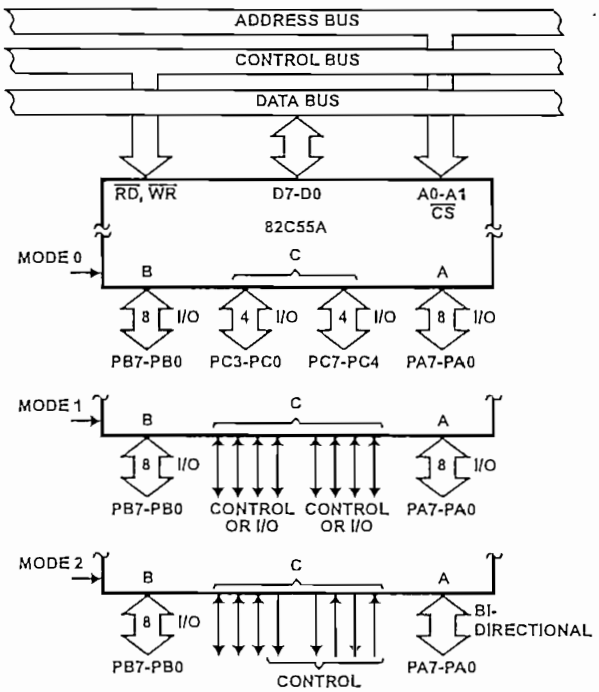


FIGURE 3. BASIC MODE DEFINITIONS AND BUS INTERFACE

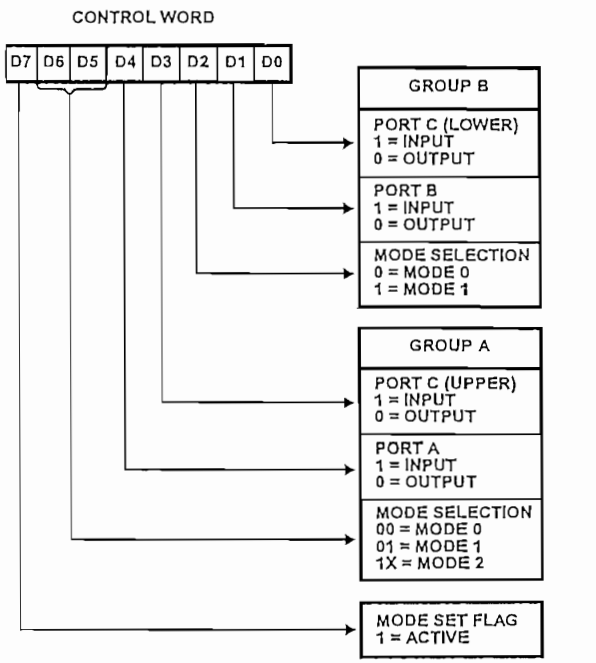


FIGURE 4. MODE DEFINITION FORMAT

The modes for Port A and Port B can be separately defined, while Port C is divided into two portions as required by the Port A and Port B definitions. All of the output registers, including the status flip-flops, will be reset whenever the mode is changed. Modes may be combined so that their functional definition can be "tailored" to almost any I/O structure. For instance: Group B can be programmed in Mode 0 to monitor simple switch closings or display computational results, Group A could be programmed in Mode 1 to monitor a keyboard or tape reader on an interrupt-driven basis.

The mode definitions and possible mode combinations may seem confusing at first, but after a cursory review of the complete device operation a simple, logical I/O approach will surface. The design of the 82C55A has taken into account things such as efficient PC board layout, control signal definition vs. PC layout and complete functional flexibility to support almost any peripheral device with no external logic. Such design represents the maximum use of the available pins.

Single Bit Set/Reset Feature (Figure 5)

Any of the eight bits of Port C can be Set or Reset using a single Output instruction. This feature reduces software requirements in control-based applications.

When Port C is being used as status/control for Port A or B, these bits can be set or reset by using the Bit Set/Reset operation just as if they were output ports.

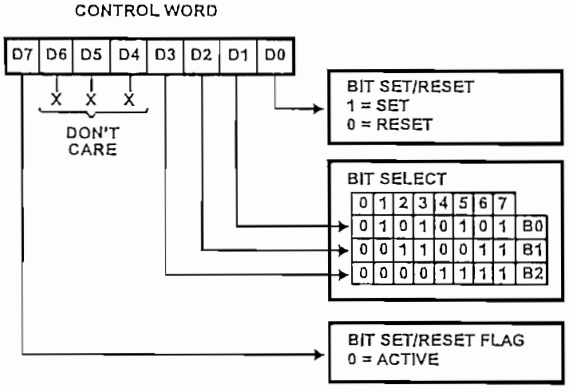


FIGURE 5. BIT SET/RESET FORMAT

Interrupt Control Functions

When the 82C55A is programmed to operate in mode 1 or mode 2, control signals are provided that can be used as interrupt request inputs to the CPU. The interrupt request signals, generated from port C, can be inhibited or enabled by setting or resetting the associated INTE flip-flop, using the bit set/reset function of port C.

This function allows the programmer to enable or disable a CPU interrupt by a specific I/O device without affecting any other device in the interrupt structure.

INTE Flip-Flop Definition

(BIT-SET)-INTE is SET - Interrupt Enable

(BIT-RESET)-INTE is Reset - Interrupt Disable

NOTE: All Mask flip-flops are automatically reset during mode selection and device Reset.

Operating Modes

Mode 0 (Basic Input/Output). This functional configuration provides simple input and output operations for each of the three ports. No handshaking is required, data is simply written to or read from a specific port.

Mode 0 Basic Functional Definitions:

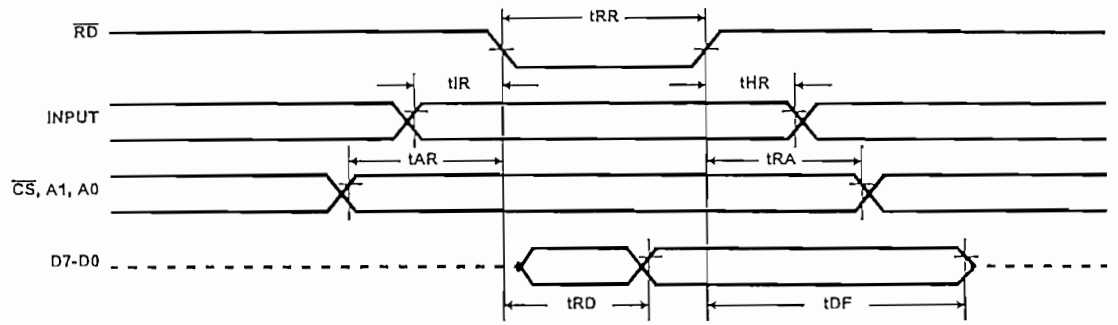
- Two 8-bit ports and two 4-bit ports
- Any Port can be input or output
- Outputs are latched
- Input are not latched
- 16 different Input/Output configurations possible

MODE 0 PORT DEFINITION

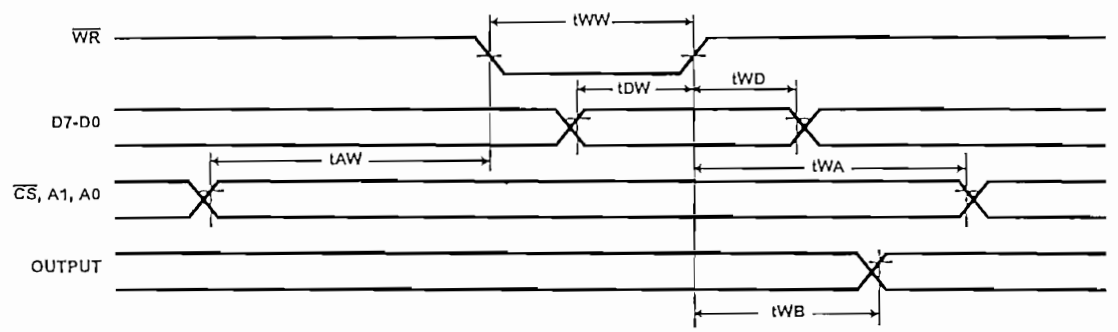
A		B		GROUP A			GROUP B	
D4	D3	D1	D0	PORT A	PORTC (Upper)	#	PORT B	PORTC (Lower)
0	0	0	0	Output	Output	0	Output	Output
0	0	0	1	Output	Output	1	Output	Input
0	0	1	0	Output	Output	2	Input	Output
0	0	1	1	Output	Output	3	Input	Input
0	1	0	0	Output	Input	4	Output	Output
0	1	0	1	Output	Input	5	Output	Input
0	1	1	0	Output	Input	6	Input	Output
0	1	1	1	Output	Input	7	Input	Input
1	0	0	0	Input	Output	8	Output	Output
1	0	0	1	Input	Output	9	Output	Input
1	0	1	0	Input	Output	10	Input	Output
1	0	1	1	Input	Output	11	Input	Input
1	1	0	0	Input	Input	12	Output	Output
1	1	0	1	Input	Input	13	Output	Input
1	1	1	0	Input	Input	14	Input	Output
1	1	1	1	Input	Input	15	Input	Input

82C55A

Mode 0 (Basic Input)



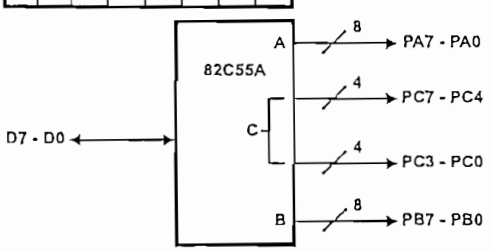
Mode 0 (Basic Output)



Mode 0 Configurations

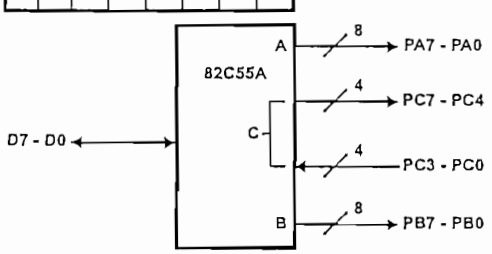
CONTROL WORD #0

D7	D6	D5	D4	D3	D2	D1	D0
1	0	0	0	0	0	0	0



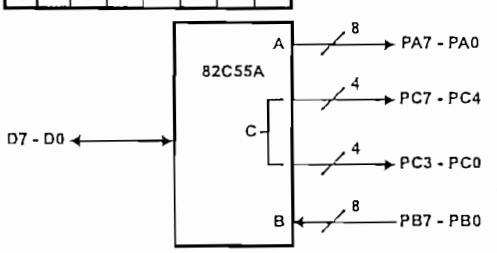
CONTROL WORD #1

D7	D6	D5	D4	D3	D2	D1	D0
1	0	0	0	0	0	0	1



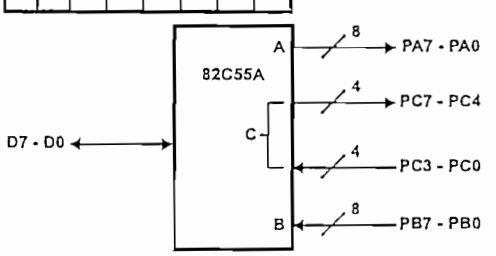
CONTROL WORD #2

D7	D6	D5	D4	D3	D2	D1	D0
1	0	0	0	0	0	1	0



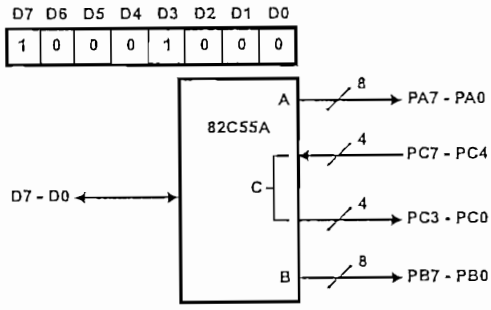
CONTROL WORD #3

D7	D6	D5	D4	D3	D2	D1	D0
1	0	0	0	0	0	1	1

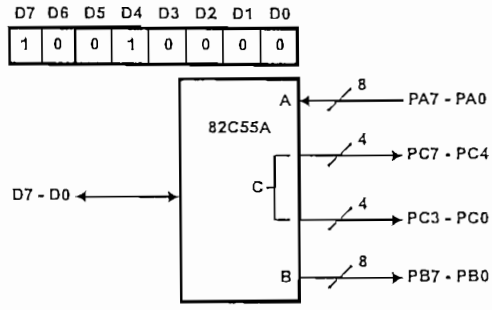


Mode 0 Configurations (Continued)

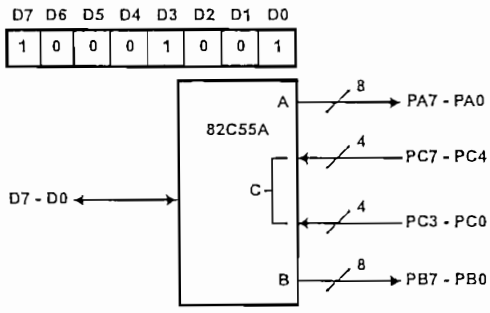
CONTROL WORD #4



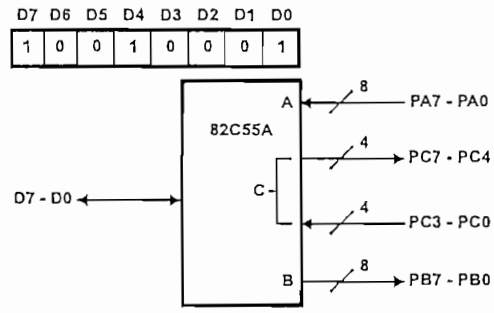
CONTROL WORD #8



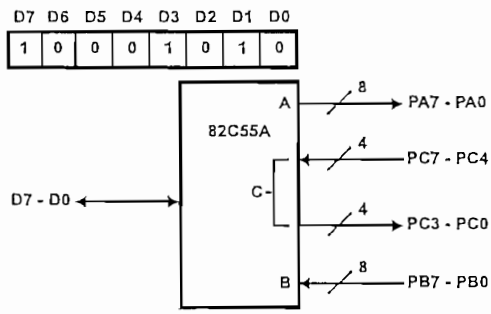
CONTROL WORD #5



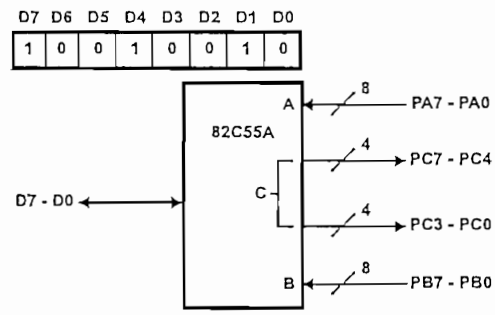
CONTROL WORD #9



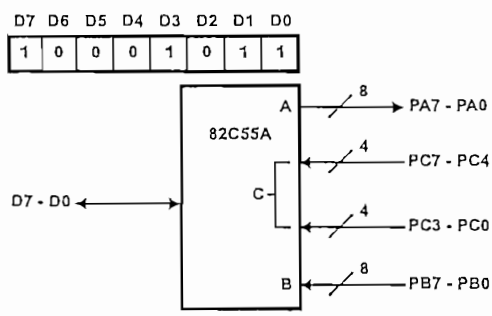
CONTROL WORD #6



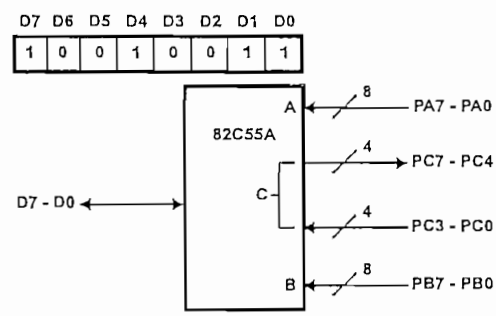
CONTROL WORD #10



CONTROL WORD #7

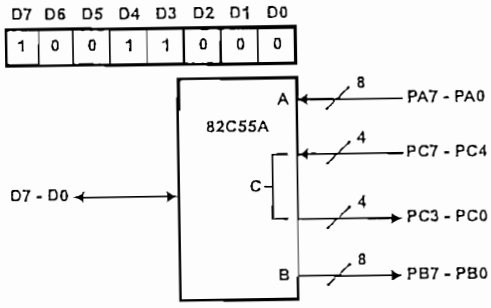


CONTROL WORD #11

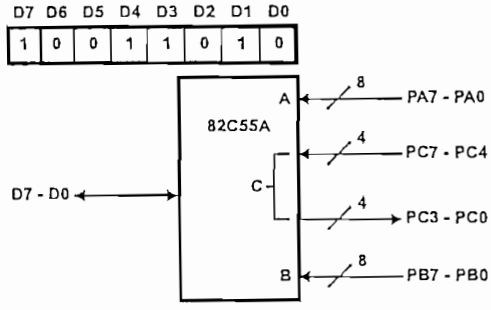


Mode 0 Configurations (Continued)

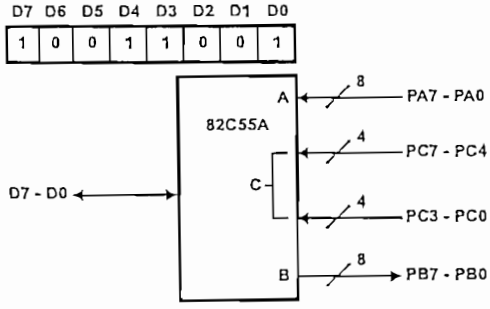
CONTROL WORD #12



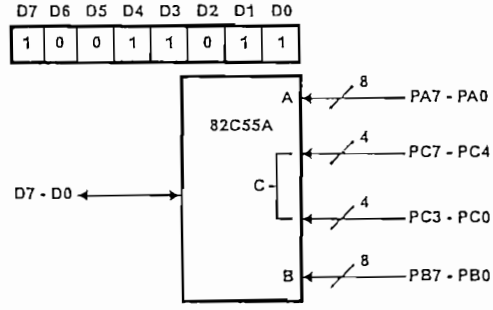
CONTROL WORD #14



CONTROL WORD #13



CONTROL WORD #15



Operating Modes

Mode 1 - (Strobed Input/Output). This functional configuration provides a means for transferring I/O data to or from a specified port in conjunction with strobes or "hand shaking" signals. In mode 1, port A and port B use the lines on port C to generate or accept these "hand shaking" signals.

Mode 1 Basic Function Definitions:

- Two Groups (Group A and Group B)
- Each group contains one 8-bit port and one 4-bit control/data port
- The 8-bit data port can be either input or output. Both inputs and outputs are latched.
- The 4-bit port is used for control and status of the 8-bit port.

Input Control Signal Definition

(Figures 6 and 7)

STB (Strobe Input)

A "low" on this input loads data into the input latch.

IBF (Input Buffer Full F/F)

A "high" on this output indicates that the data has been loaded into the input latch: in essence, and acknowledgment. IBF is set by $\overline{STB}$ input being low and is reset by the rising edge of the $\overline{RD}$ input.

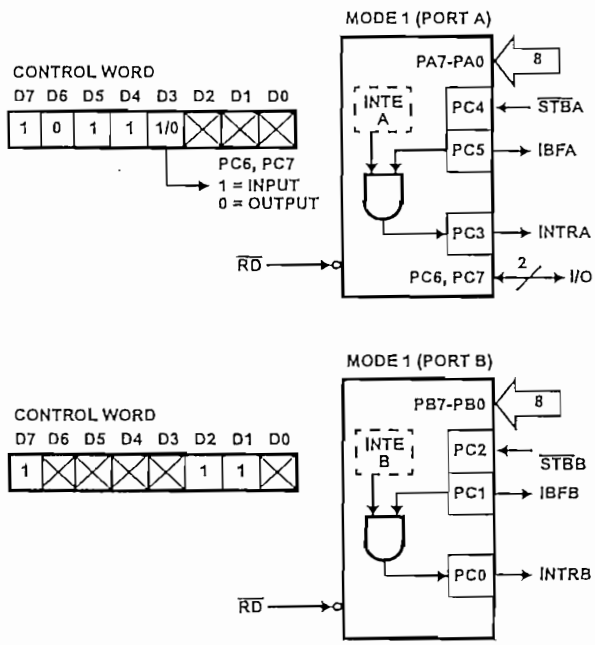


FIGURE 6. MODE 1 INPUT

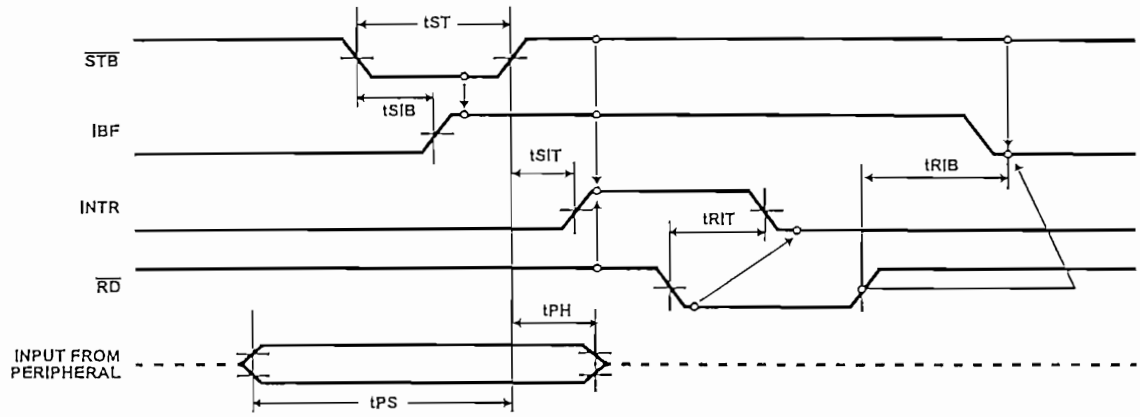


FIGURE 7. MODE 1 (STROBED INPUT)

INTR (Interrupt Request)

A "high" on this output can be used to interrupt the CPU when and input device is requesting service. INTR is set by the condition: $\overline{STB}$ is a "one", IBF is a "one" and INTE is a "one". It is reset by the falling edge of $\overline{RD}$. This procedure allows an input device to request service from the CPU by simply strobing its data into the port.

INTE A

Controlled by bit set/reset of PC4.

INTE B

Controlled by bit set/reset of PC2.

Output Control Signal Definition

(Figure 8 and 9)

$\overline{OBF}$ - Output Buffer Full F/F). The $\overline{OBF}$ output will go "low" to indicate that the CPU has written data out to be specified port. This does not mean valid data is sent out of the part at this time since $\overline{OBF}$ can go true before data is available. Data is guaranteed valid at the rising edge of $\overline{OBF}$, (See Note 1). The $\overline{OBF}$ F/F will be set by the rising edge of the $\overline{WR}$ input and reset by $\overline{ACK}$ input being low.

$\overline{ACK}$ - Acknowledge Input). A "low" on this input informs the 82C55A that the data from Port A or Port B is ready to be accepted. In essence, a response from the peripheral device indicating that it is ready to accept data, (See Note 1).

INTR - (Interrupt Request). A "high" on this output can be used to interrupt the CPU when an output device has accepted data transmitted by the CPU. INTR is set when $\overline{ACK}$ is a "one", OBF is a "one" and INTE is a "one". It is reset by the falling edge of $\overline{WR}$.

INTE A

Controlled by Bit Set/Reset of PC6.

INTE B

Controlled by Bit Set/Reset of PC2.

NOTE:

1. To strobe data into the peripheral device, the user must operate the strobe line in a hand shaking mode. The user needs to send $\overline{OBF}$ to the peripheral device, generates an $\overline{ACK}$ from the peripheral device and then latch data into the peripheral device on the rising edge of $\overline{OBF}$.

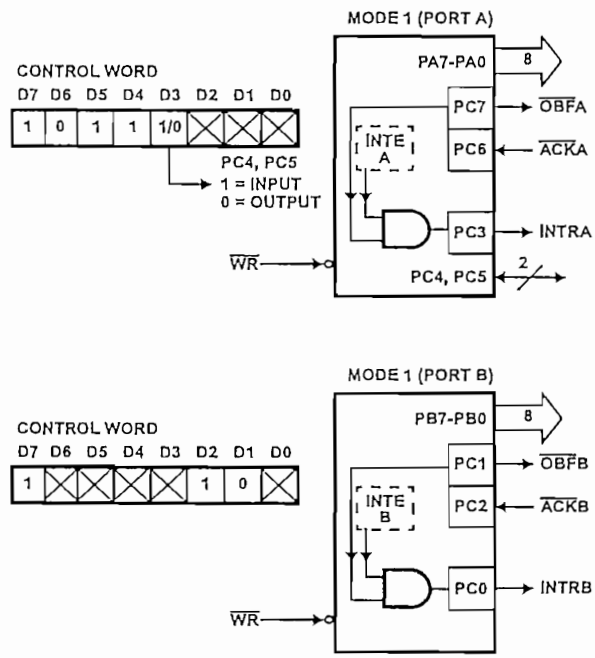


FIGURE 8. MODE 1 OUTPUT

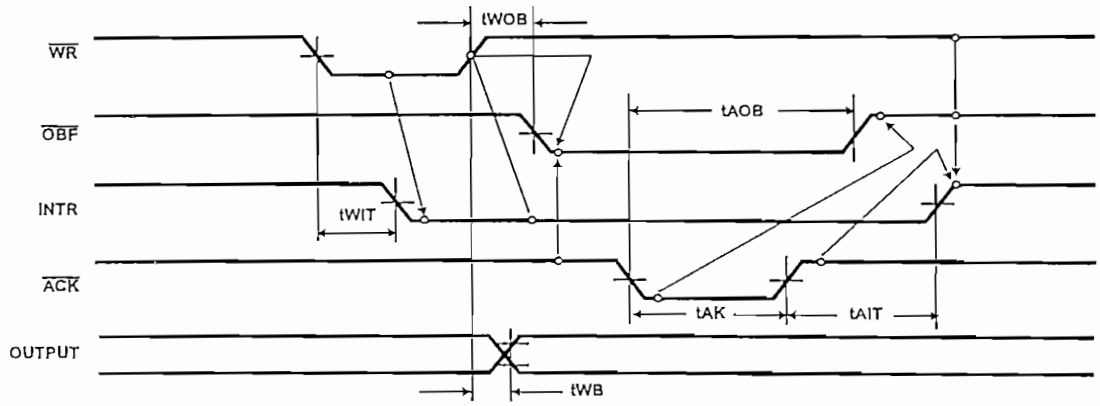
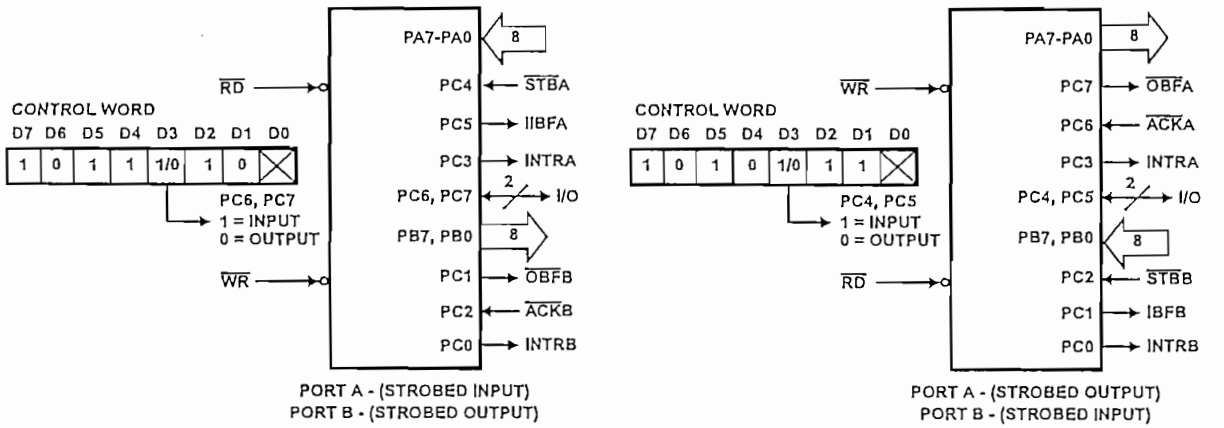


FIGURE 9. MODE 1 (STROBED OUTPUT)



Combinations of Mode 1: Port A and Port B can be individually defined as input or output in Mode 1 to support a wide variety of strobed I/O applications.

FIGURE 10. COMBINATIONS OF MODE 1

Operating Modes

Mode 2 (Strobed Bi-Directional Bus I/O)

The functional configuration provides a means for communicating with a peripheral device or structure on a single 8-bit bus for both transmitting and receiving data (bi-directional bus I/O). "Hand shaking" signals are provided to maintain proper bus flow discipline similar to Mode 1. Interrupt generation and enable/disable functions are also available.

Mode 2 Basic Functional Definitions:

- Used in Group A only
- One 8-bit, bi-directional bus Port (Port A) and a 5-bit control Port (Port C)
- Both inputs and outputs are latched
- The 5-bit control port (Port C) is used for control and status for the 8-bit, bi-directional bus port (Port A)

Bi-Directional Bus I/O Control Signal Definition

(Figures 11, 12, 13, 14)

INTR - (Interrupt Request). A high on this output can be used to interrupt the CPU for both input or output operations.

Output Operations

$\overline{OBF}$ - (Output Buffer Full). The $\overline{OBF}$ output will go "low" to indicate that the CPU has written data out to port A.

$\overline{ACK}$ - (Acknowledge). A "low" on this input enables the three-state output buffer of port A to send out the data. Otherwise, the output buffer will be in the high impedance state.

INTE 1 - (The INTE flip-flop associated with $\overline{OBF}$). Controlled by bit set/reset of PC4.

Input Operations

$\overline{STB}$ - (Strobe Input). A "low" on this input loads data into the input latch.

IBF - (Input Buffer Full F/F). A "high" on this output indicates that data has been loaded into the input latch.

INTE 2 - (The INTE flip-flop associated with IBF). Controlled by bit set/reset of PC4.

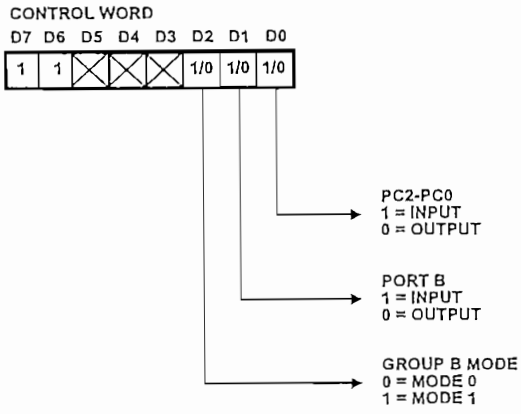


FIGURE 11. MODE CONTROL WORD

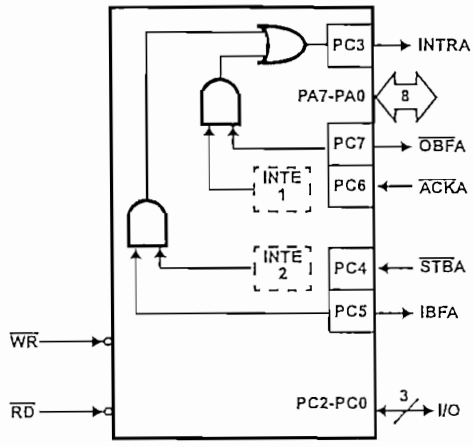
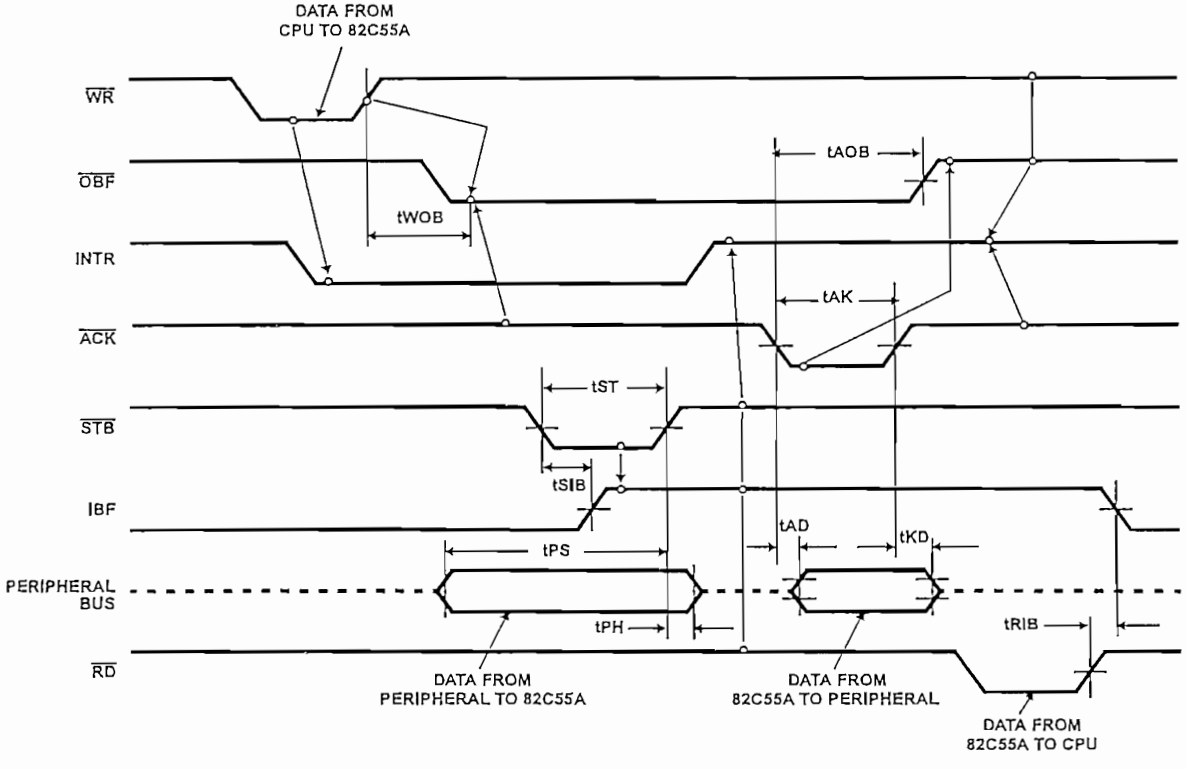


FIGURE 12. MODE 2



NOTE: Any sequence where $\overline{WR}$ occurs before $\overline{ACK}$ and $\overline{STB}$ occurs before RD is permissible. (INTR = IBF • MASK • $\overline{STB}$ • $\overline{RD}$ + $\overline{OBF}$ • MASK • $\overline{ACK}$ • $\overline{WR}$)

FIGURE 13. MODE 2 (BI-DIRECTIONAL)

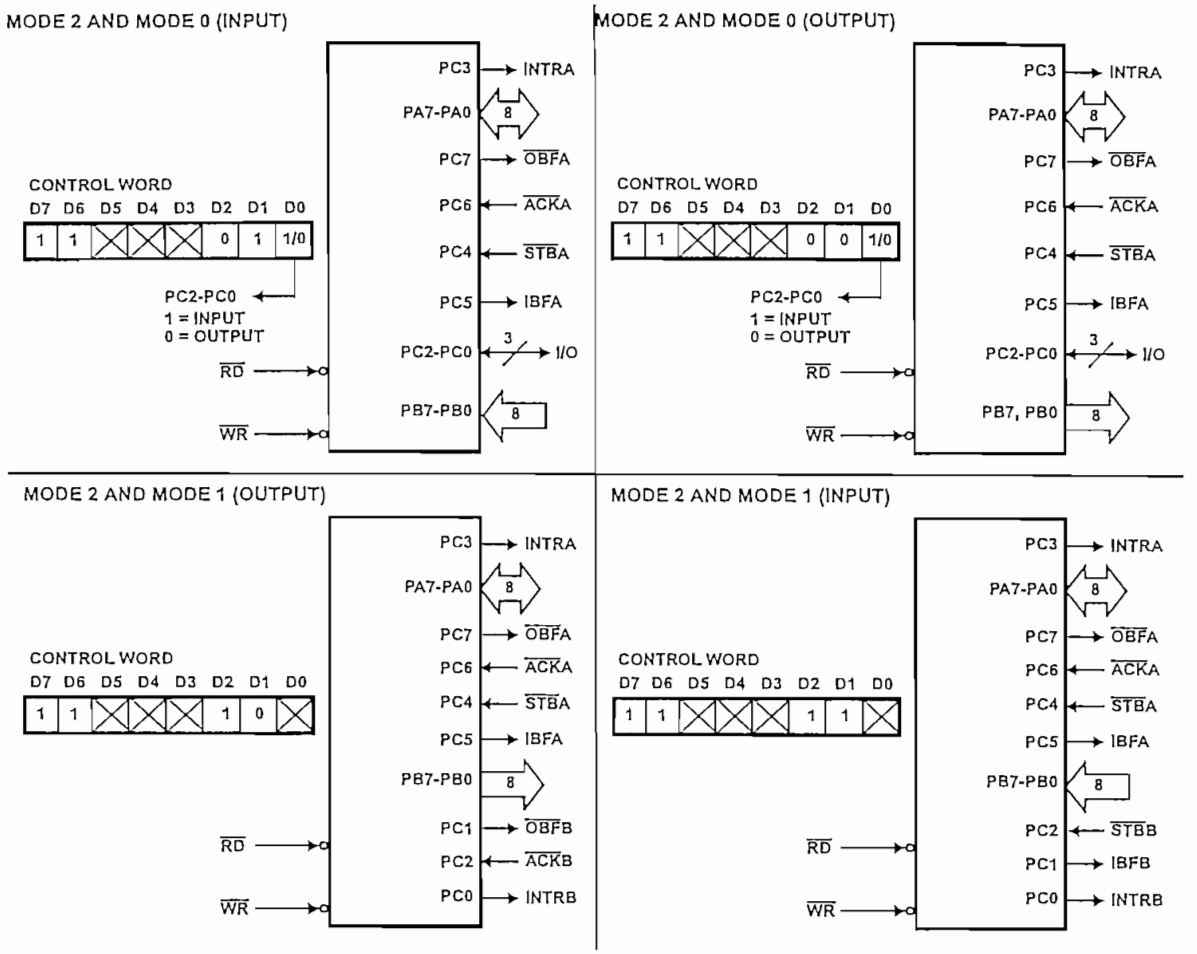


FIGURE 14. MODE 2 COMBINATIONS

MODE DEFINITION SUMMARY

	MODE 0		MODE 1		MODE 2	
	IN	OUT	IN	OUT	GROUP A ONLY	
PA0	In	Out	In	Out	← →	
PA1	In	Out	In	Out	← →	
PA2	In	Out	In	Out	← →	
PA3	In	Out	In	Out	← →	
PA4	In	Out	In	Out	← →	
PA5	In	Out	In	Out	← →	
PA6	In	Out	In	Out	← →	
PA7	In	Out	In	Out	← →	
PB0	In	Out	In	Out		Mode 0 or Mode 1 Only
PB1	In	Out	In	Out		
PB2	In	Out	In	Out		
PB3	In	Out	In	Out		
PB4	In	Out	In	Out		
PB5	In	Out	In	Out		
PB6	In	Out	In	Out		
PB7	In	Out	In	Out		
PC0	In	Out	INTRB	INTRB	I/O	
PC1	In	Out	IBFB	$\overline{OBF}$B	I/O	
PC2	In	Out	$\overline{STB}$B	$\overline{ACK}$B	I/O	
PC3	In	Out	INTRA	INTRA	INTRA	
PC4	In	Out	$\overline{STB}$A	I/O	$\overline{STB}$A	
PC5	In	Out	IBFA	I/O	IBFA	
PC6	In	Out	I/O	$\overline{ACK}$A	$\overline{ACK}$A	
PC7	In	Out	I/O	$\overline{OBF}$A	$\overline{OBF}$A	

Special Mode Combination Considerations

There are several combinations of modes possible. For any combination, some or all of Port C lines are used for control or status. The remaining bits are either inputs or outputs as defined by a "Set Mode" command.

During a read of Port C, the state of all the Port C lines, except the $\overline{ACK}$ and $\overline{STB}$ lines, will be placed on the data bus. In place of the $\overline{ACK}$ and $\overline{STB}$ line states, flag status will appear on the data bus in the PC2, PC4, and PC6 bit *positions as illustrated* by Figure 17.

Through a "Write Port C" command, only the Port C pins programmed as outputs in a Mode 0 group can be written. No other pins can be affected by a "Write Port C" command, nor can the interrupt enable flags be accessed. To write to any Port C output programmed as an output in Mode 1 group or to change an interrupt enable flag, the "Set/Reset Port C Bit" command must be used.

With a "Set/Reset Port Cea Bit" command, any Port C line programmed as an output (including IBF and $\overline{OBF}$) can be written, or an interrupt enable flag can be either set or reset. Port C lines programmed as inputs, including $\overline{ACK}$ and $\overline{STB}$ lines, associated with Port C fare not affected by a "Set/Reset Port C Bit" command. Writing to the corresponding Port C bit positions of the $\overline{ACK}$ and $\overline{STB}$ lines with the "Set Reset Port C Bit" command will affect the Group A and Group B interrupt enable flags, as illustrated in Figure 17.

INPUT CONFIGURATION

D7	D6	D5	D4	D3	D2	D1	D0
I/O	I/O	IBFA	INTEA	INTRA	INTEB	IBFB	INTRB
GROUP A					GROUP B		

OUTPUT CONFIGURATION

D7	D6	D5	D4	D3	D2	D1	D0
OBFA	INTEA	I/O	I/O	INTRA	INTEB	OBFB	INTRB
GROUP A					GROUP B		

FIGURE 15. MODE 1 STATUS WORD FORMAT

D7	D6	D5	D4	D3	D2	D1	D0
OBFA	INTE1	IBFA	INTE2	INTRA	X	X	X
GROUP A					GROUP B		

(Defined by Mode 0 or Mode 1 Selection)

FIGURE 16. MODE 2 STATUS WORD FORMAT

Current Drive Capability

Any output on Port A, B or C can sink or source 2.5mA. This feature allows the 82C55A to directly drive Darlington type drivers and high-voltage displays that require such sink or source current.

Reading Port C Status (Figures 15 and 16)

In Mode 0, Port C transfers data to or from the peripheral device. When the 82C55A is programmed to function in Modes 1 or 2, Port C generates or accepts "hand shaking" signals with the peripheral device. Reading the contents of Port C allows the programmer to test or verify the "status" of each peripheral device and change the program flow accordingly.

There is not special instruction to read the status information from Port C. A normal read operation of Port C is executed to perform this function.

INTERRUPT ENABLE FLAG	POSITION	ALTERNATE PORT C PIN SIGNAL (MODE)
INTE B	PC2	$\overline{ACK}$B (Output Mode 1) or $\overline{STB}$B (Input Mode 1)
INTE A2	PC4	$\overline{STB}$A (Input Mode 1 or Mode 2)
INTE A1	PC6	$\overline{ACK}$A (Output Mode 1 or Mode 2)

FIGURE 17. INTERRUPT ENABLE FLAGS IN MODES 1 AND 2

Applications of the 82C55A

The 82C55A is a very powerful tool for interfacing peripheral equipment to the microcomputer system. It represents the optimum use of available pins and flexible enough to interface almost any I/O device without the need for additional external logic.

Each peripheral device in a microcomputer system usually has a "service routine" associated with it. The routine manages the software interface between the device and the CPU. The functional definition of the 82C55A is programmed by the I/O service routine and becomes an extension of the system software. By examining the I/O devices interface characteristics for both data transfer and timing, and matching this information to the examples and tables in the detailed operational description, a control word can easily be developed to initialize the 82C55A to exactly "fit" the application. Figures 18 through 24 present a few examples of typical applications of the 82C55A.

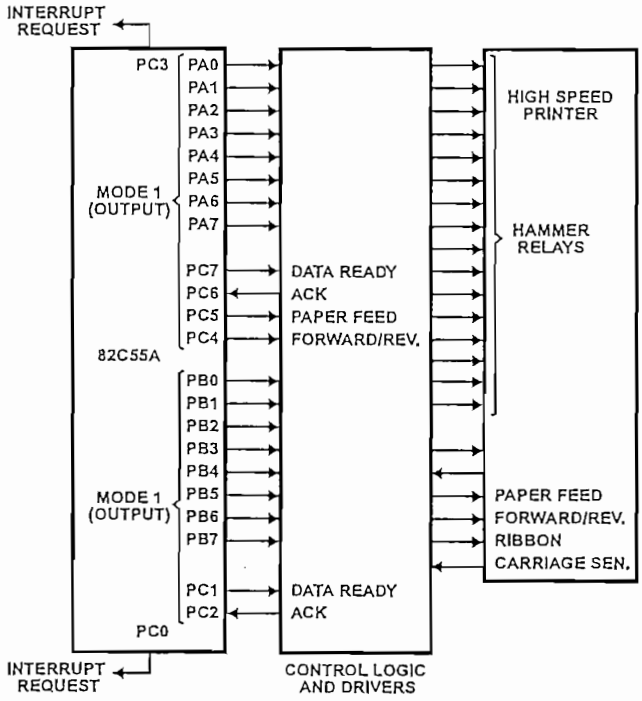


FIGURE 18. PRINTER INTERFACE

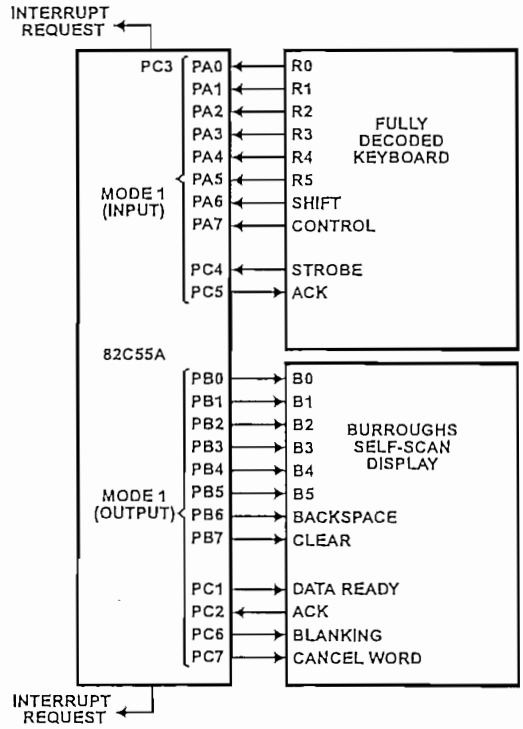


FIGURE 19. KEYBOARD AND DISPLAY INTERFACE

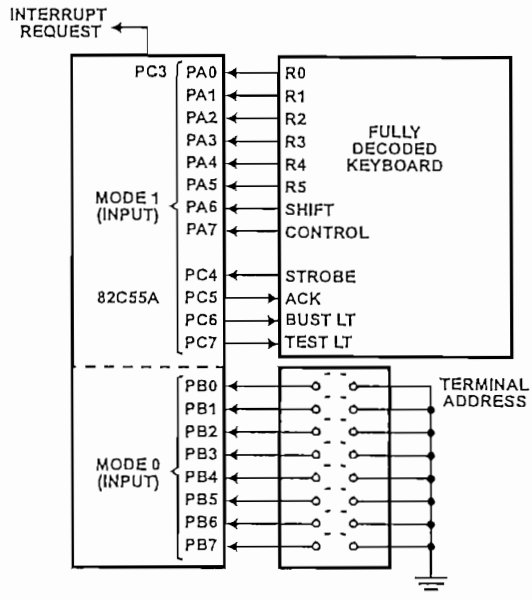


FIGURE 20. KEYBOARD AND TERMINAL ADDRESS INTERFACE

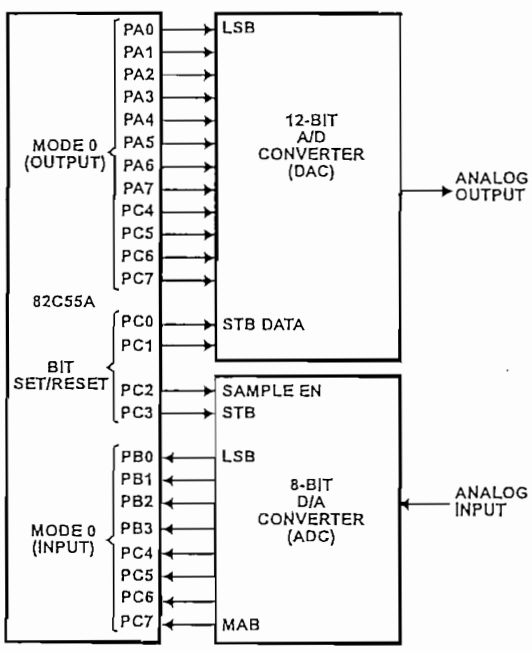


FIGURE 21. DIGITAL TO ANALOG, ANALOG TO DIGITAL

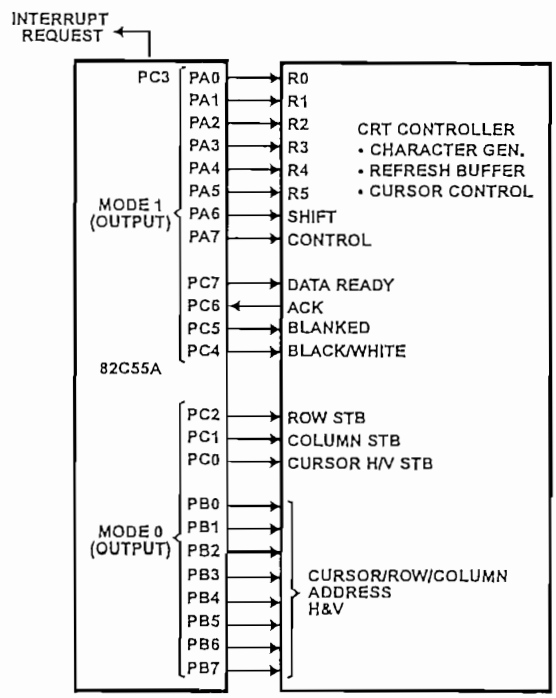


FIGURE 22. BASIC CRT CONTROLLER INTERFACE

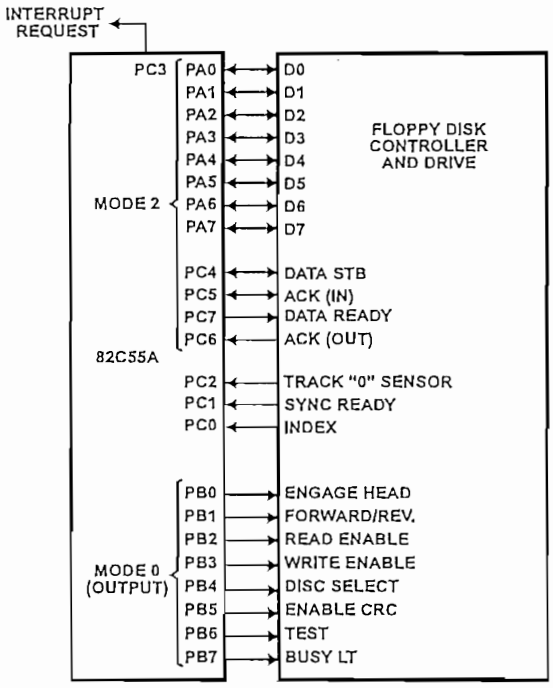


FIGURE 23. BASIC FLOPPY DISC INTERFACE

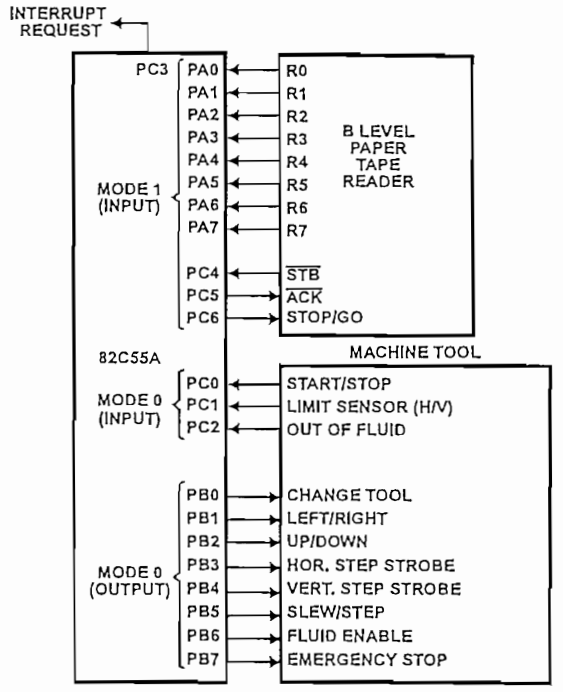


FIGURE 24. MACHINE TOOL CONTROLLER INTERFACE

Absolute Maximum Ratings $T_A = 25^oC$

Supply Voltage ..... +8.0V
Input, Output or I/O Voltage ..... GND-0.5V to V_{CC}+0.5V
ESD Classification ..... Class 1

Operating Conditions

Voltage Range ..... +4.5V to 5.5V
Operating Temperature Range
C82C55A ..... 0^oC to 70^oC
I82C55A ..... -40^oC to 85^oC
M82C55A ..... -55^oC to 125^oC

Thermal Information

Thermal Resistance (Typical, Note 1)	θ_{JA}	θ_{JC}
CERDIP Package	$50^oC/W$	$10^oC/W$
CLCC Package	$65^oC/W$	$14^oC/W$
PDIP Package	$50^oC/W$	N/A
PLCC Package	$46^oC/W$	N/A

Maximum Storage Temperature Range ..... -65^oC to 150^oC
Maximum Junction Temperature
CDIP Package ..... 175^oC
PDIP Package ..... 150^oC
Maximum Lead Temperature (Soldering 10s) ..... 300^oC
(PLCC Lead Tips Only)

Die Characteristics

Gate Count ..... 1000 Gates

CAUTION: Stresses above those listed in "Absolute Maximum Ratings" may cause permanent damage to the device. This is a stress only rating and operation of the device at these or any other conditions above those indicated in the operational sections of this specification is not implied.

NOTE:

1. θ_{JA} is measured with the component mounted on an evaluation PC board in free air.

Electrical Specifications

V_{CC} = 5.0V ±10%; $T_A = 0^oC$ to $+70^oC$ (C82C55A);
$T_A = -40^oC$ to $+85^oC$ (I82C55A);
$T_A = -55^oC$ to $+125^oC$ (M82C55A)

SYMBOL	PARAMETER	LIMITS MIN	LIMITS MAX	UNITS	TEST CONDITIONS
V_{IH}	Logical One Input Voltage	2.0 2.2	-	V	I82C55A, C82C55A, M82C55A
V_{IL}	Logical Zero Input Voltage	-	0.8	V	
V_{OH}	Logical One Output Voltage	3.0 V_{CC} -0.4	-	V	I_{OH} = -2.5mA, I_{OH} = -100µA
V_{OL}	Logical Zero Output Voltage	-	0.4	V	I_{OL} +2.5mA
I_I	Input Leakage Current	-1.0	+1.0	µA	$V_{IN} = V_{CC}$ or GND, DIP Pins: 5, 6, 8, 9, 35, 36
IO	I/O Pin Leakage Current	-10	+10	µA	VO = V_{CC} or GND DIP Pins: 27 - 34
IBHH	Bus Hold High Current	-50	-400	µA	VO = 3.0V. Ports A, B, C
IBHL	Bus Hold Low Current	50	400	µA	VO = 1.0V. Port A ONLY
IDAR	Darlington Drive Current	-2.5	Note 2, 4	mA	Ports A, B, C. Test Condition 3
ICCSB	Standby Power Supply Current	-	10	µA	V_{CC} = 5.5V, $V_{IN} = V_{CC}$ or GND. Output Open
ICCOP	Operating Power Supply Current	-	1	mA/MHz	$T_A = +25^oC$, V_{CC} = 5.0V, Typical (See Note 3)

NOTES:

2. No internal current limiting exists on Port Outputs. A resistor must be added externally to limit the current.
3. ICCOP = 1mA/MHz of Peripheral Read/Write cycle time. (Example: 1.0µs I/O Read/Write cycle time = 1mA).
4. Tested as V_{OH} at -2.5mA.

Capacitance $T_A = 25^oC$

SYMBOL	PARAMETER	TYPICAL	UNITS	TEST CONDITIONS
CIN	Input Capacitance	10	pF	FREQ = 1MHz, All Measurements are referenced to device GND
CI/O	I/O Capacitance	20	pF	

AC Electrical Specifications

V_{CC} = +5V± 10%, GND = 0V; T_A = -55°C to +125°C (M82C55A) (M82C55A-5);
T_A = -40°C to +85°C (I82C55A) (I82C55A-5);
T_A = 0°C to +70°C (C82C55A) (C82C55A-5)

SYMBOL	PARAMETER	82C55A-5		82C55A		UNITS	TEST CONDITIONS
		MIN	MAX	MIN	MAX		
READ TIMING							
(1) tAR	Address Stable Before $\overline{RD}$	0	-	0	-	ns	
(2) tRA	Address Stable After $\overline{RD}$	0	-	0	-	ns	
(3) tRR	$\overline{RD}$ Pulse Width	250	-	150	-	ns	
(4) tRD	Data Valid From $\overline{RD}$	-	200	-	120	ns	1
(5) tDF	Data Float After $\overline{RD}$	10	75	10	75	ns	2
(6) tRV	Time Between $\overline{RD}$s and/or $\overline{WR}$s	300	-	300	-	ns	
WRITE TIMING							
(7) tAW	Address Stable Before $\overline{WR}$	0	-	0	-	ns	
(8) tWA	Address Stable After $\overline{WR}$	20	-	20	-	ns	
(9) tWW	$\overline{WR}$ Pulse Width	100	-	100	-	ns	
(10) tDW	Data Valid to $\overline{WR}$ High	100	-	100	-	ns	
(11) tWD	Data Valid After $\overline{WR}$ High	30	-	30	-	ns	
OTHER TIMING							
(12) tWB	$\overline{WR}$ = 1 to Output	-	350	-	350	ns	1
(13) tIR	Peripheral Data Before $\overline{RD}$	0	-	0	-	ns	
(14) tHR	Peripheral Data After $\overline{RD}$	0	-	0	-	ns	
(15) tAK	ACK Pulse Width	200	-	200	-	ns	
(16) tST	STB Pulse Width	100	-	100	-	ns	
(17) tPS	Peripheral Data Before STB High	20	-	20	-	ns	
(18) tPH	Peripheral Data After STB High	50	-	50	-	ns	
(19) tAD	ACK = 0 to Output	-	175	-	175	ns	1
(20) tKD	ACK = 1 to Output Float	20	250	20	250	ns	2
(21) tWOB	$\overline{WR}$ = 1 to OBF = 0	-	150	-	150	ns	1
(22) tAOB	ACK = 0 to OBF = 1	-	150	-	150	ns	1
(23) tSIB	STB = 0 to IBF = 1	-	150	-	150	ns	1
(24) tRIB	$\overline{RD}$ = 1 to IBF = 0	-	150	-	150	ns	1
(25) tRIT	$\overline{RD}$ = 0 to INTR = 0	-	200	-	200	ns	1
(26) tSIT	STB = 1 to INTR = 1	-	150	-	150	ns	1
(27) tAIT	ACK = 1 to INTR = 1	-	150	-	150	ns	1
(28) tWIT	$\overline{WR}$ = 0 to INTR = 0	-	200	-	200	ns	1
(29) tRES	Reset Pulse Width	500	-	500	-	ns	1, (Note)

NOTE: Period of initial Reset pulse after power-on must be at least 50µsec. Subsequent Reset pulses may be 500ns minimum.

Timing Waveforms

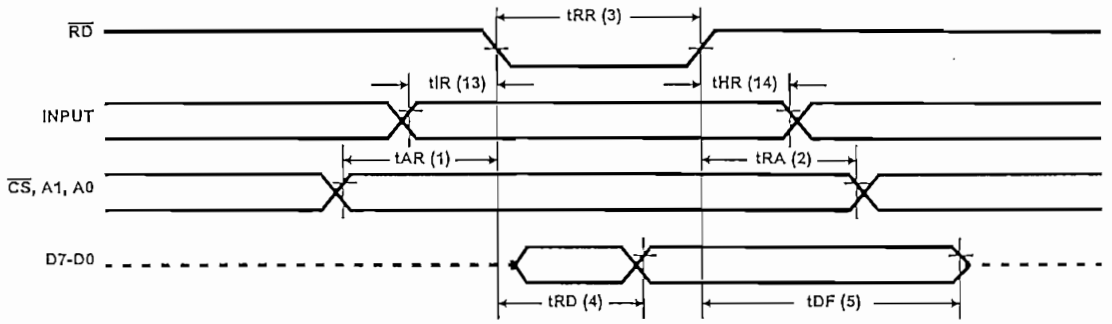


FIGURE 25. MODE 0 (BASIC INPUT)

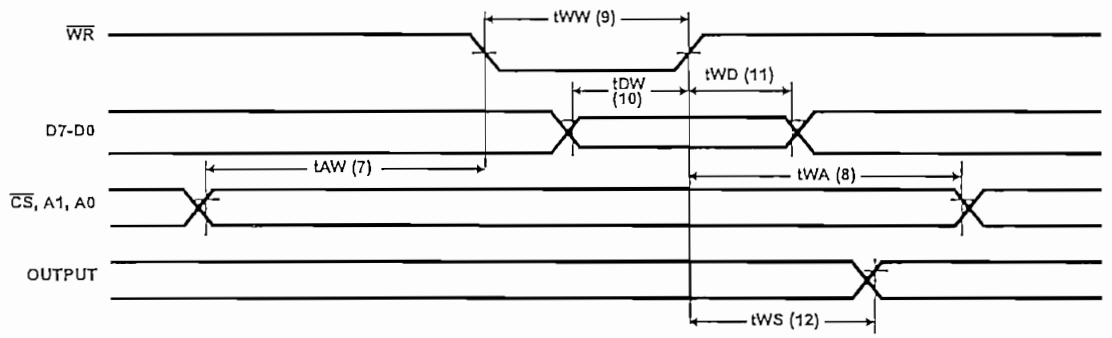


FIGURE 26. MODE 0 (BASIC OUTPUT)

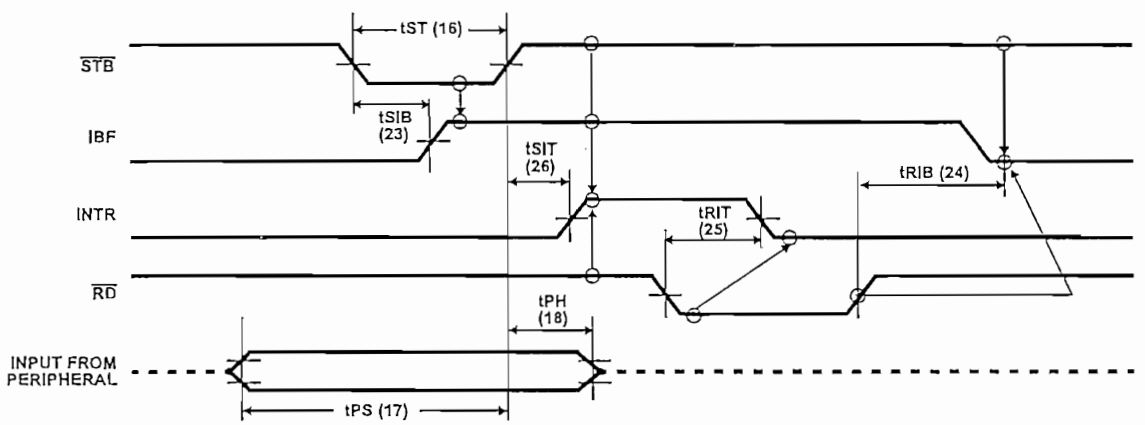


FIGURE 27. MODE 1 (STROBED INPUT)

Timing Waveforms (Continued)

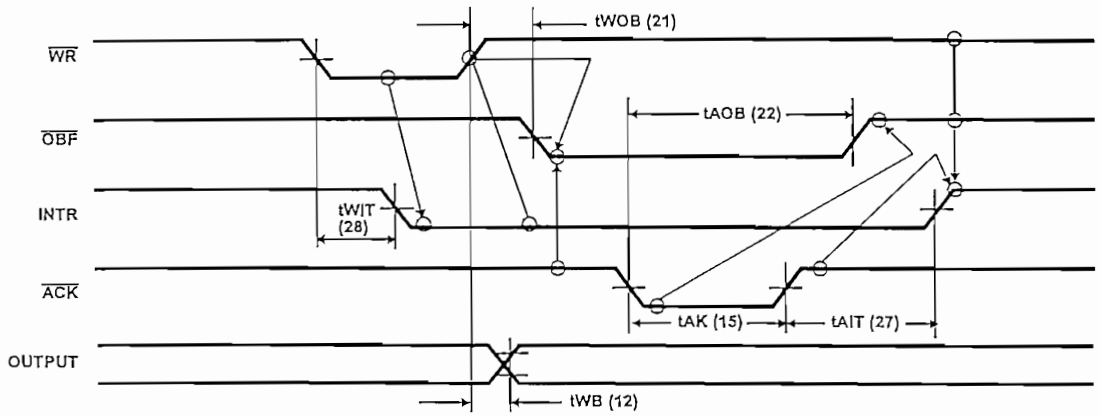


FIGURE 28. MODE 1 (STROBED OUTPUT)

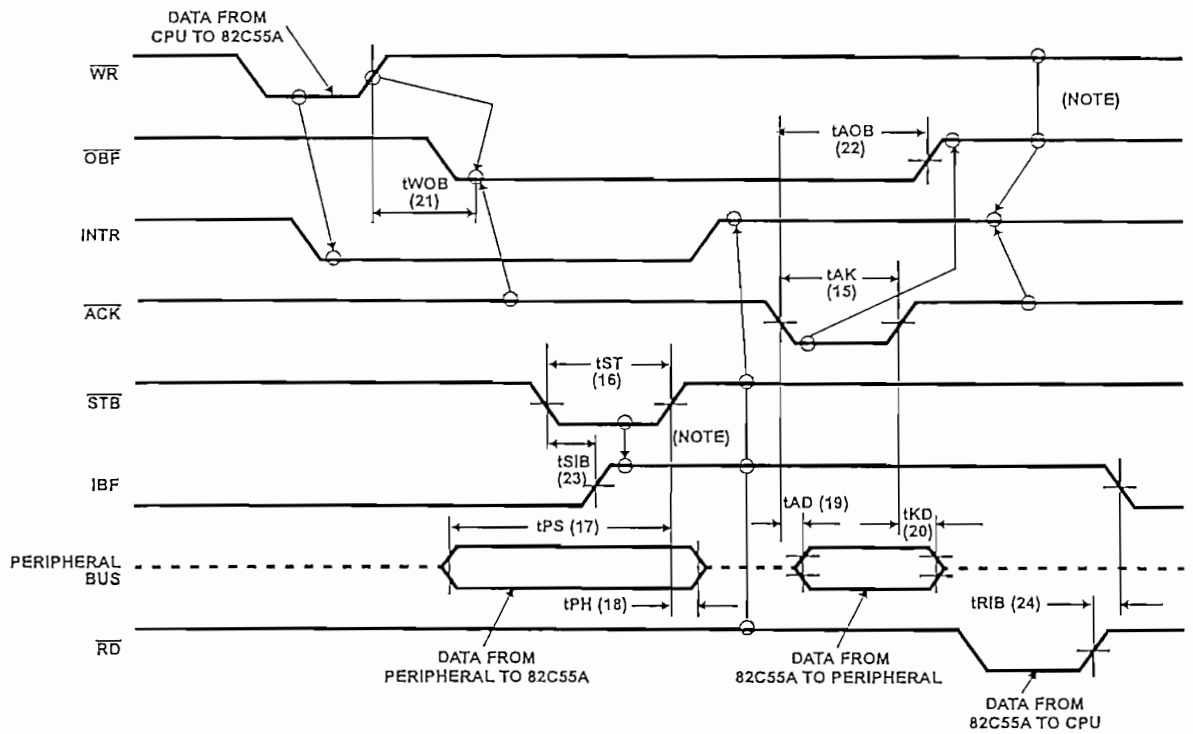


FIGURE 29. MODE 2 (BI-DIRECTIONAL)

NOTE: Any sequence where $\overline{WR}$ occurs before $\overline{ACK}$ and $\overline{STB}$ occurs before $\overline{RD}$ is permissible. (INTR = IBF • $\overline{MASK}$ • $\overline{STB}$ • $\overline{RD}$ • $\overline{OBF}$ • $\overline{MASK}$ • $\overline{ACK}$ • $\overline{WR}$)

Timing Waveforms (Continued)

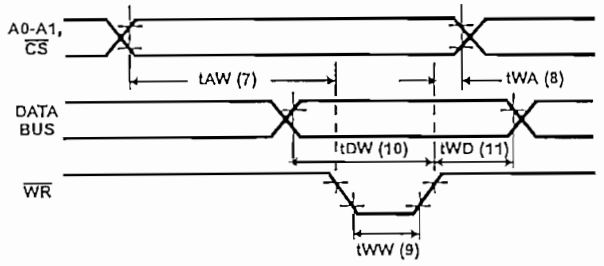


FIGURE 30. WRITE TIMING

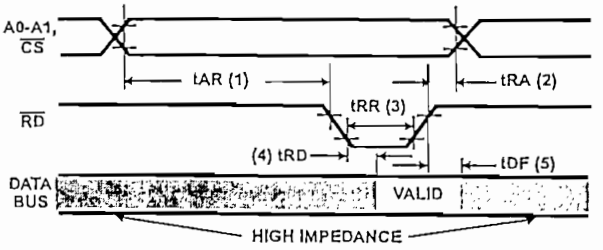
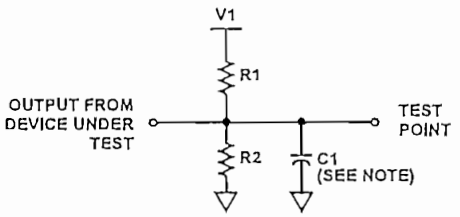


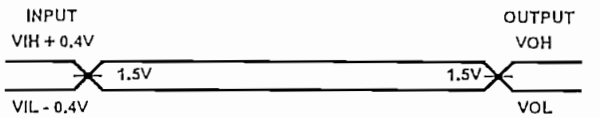
FIGURE 31. READ TIMING

AC Test Circuit



NOTE: Includes STRAY and JIG Capacitance

AC Testing Input, Output Waveforms



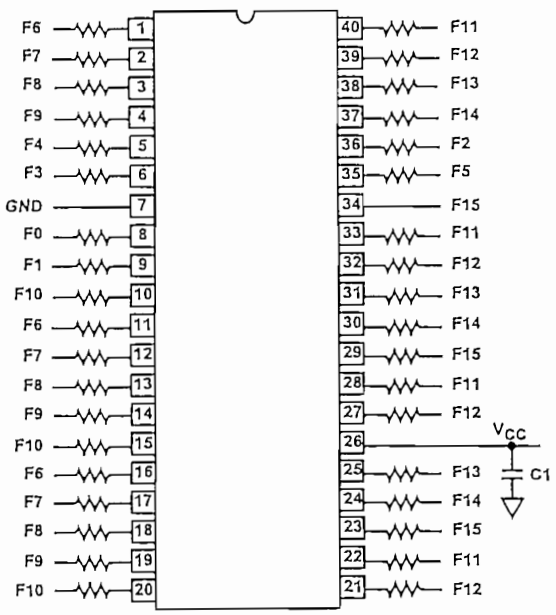
AC Testing: All AC Parameters tested as per test circuits. Input RISE and FALL times are driven at 1ns/V.

TEST CONDITION DEFINITION TABLE

TEST CONDITION	V1	R1	R2	C1
1	1.7V	523Ω	Open	150pF
2	V_{CC}	2kΩ	1.7kΩ	50pF
3	1.5V	750Ω	Open	50pF

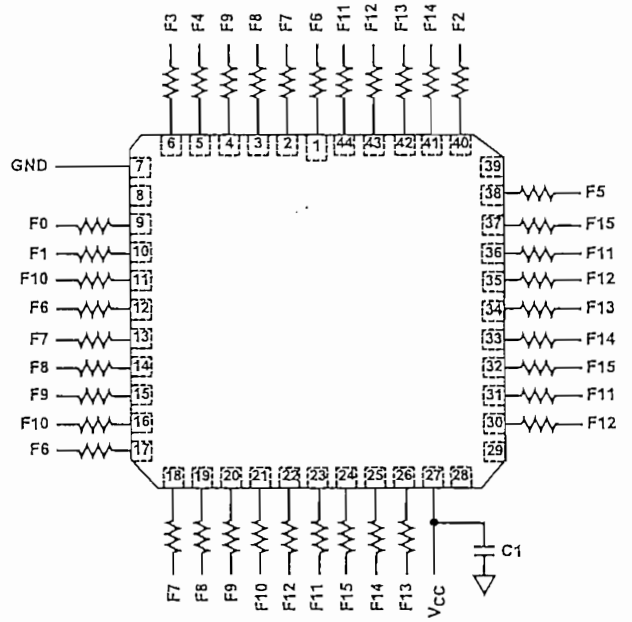
Burn-In Circuits

MD82C55A CERDIP



NOTES:
1. V_{CC} = 5.5V ± 0.5V
2. VIH = 4.5V ± 10%
3. VIL = -0.2V to 0.4V
4. GND = 0V

MR82C55A CLCC



NOTES:
1. C1 = 0.01µF minimum
2. All resistors are 47kΩ ± 5%
3. f0 = 100kHz ± 10%
4. f1 = f0 ÷ 2; f2 = f1 ÷ 2; . . . ; f15 = f14 ÷ 2

Die Characteristics

DIE DIMENSIONS:
95 x 100 x 19 ±1mils

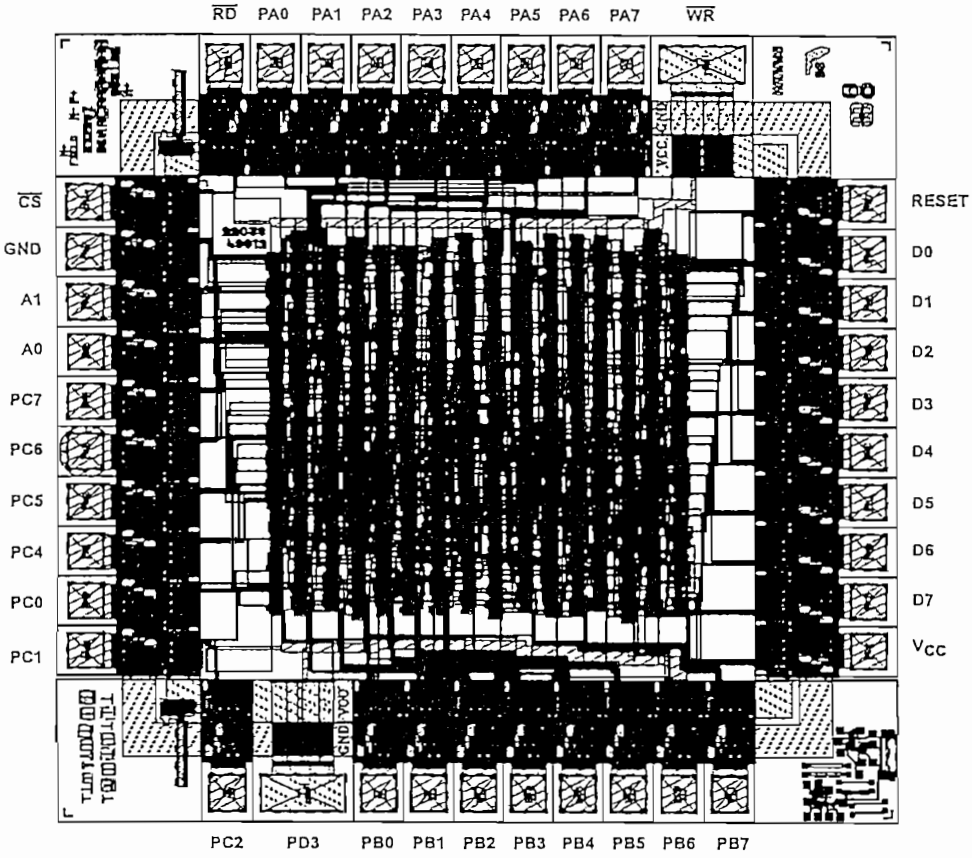
METALLIZATION:
Type: Silicon - Aluminum
Thickness: 11kÅ ±1kÅ

GLASSIVATION:
Type: SiO_2
Thickness: 8kÅ ±1kÅ

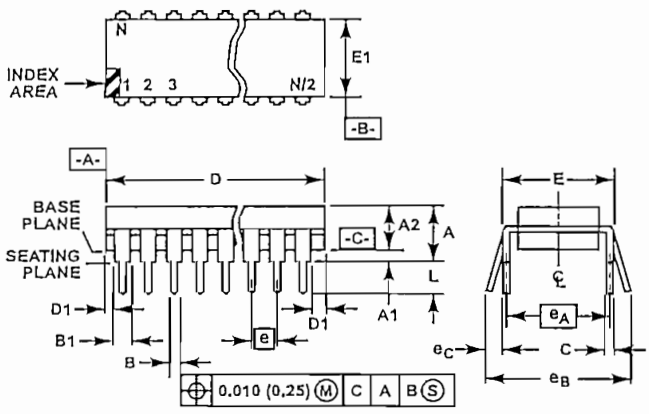
WORST CASE CURRENT DENSITY:
0.78×10^5 A/cm^2

Metallization Mask Layout

82C55A



Dual-In-Line Plastic Packages (PDIP)



E40.6 (JEDEC MS-011-AC ISSUE B)
40 LEAD DUAL-IN-LINE PLASTIC PACKAGE

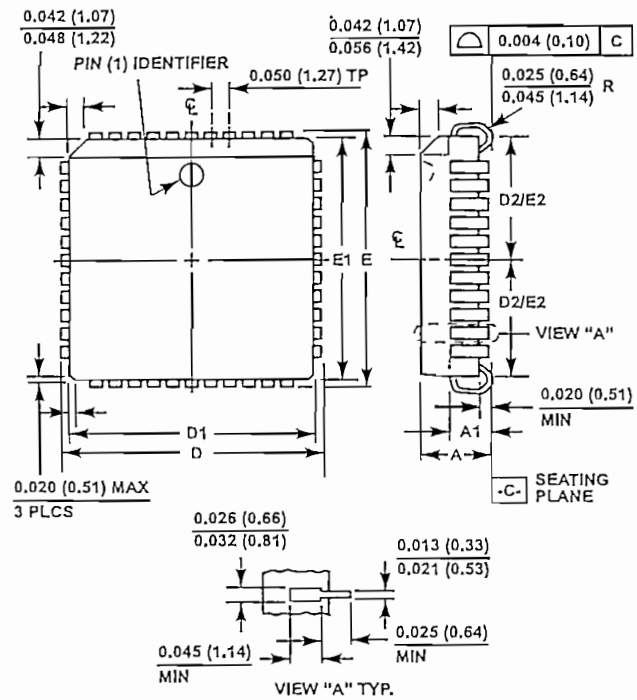
SYMBOL	INCHES		MILLIMETERS		NOTES
	MIN	MAX	MIN	MAX	
A	-	0.250	-	6.35	4
A1	0.015	-	0.39	-	4
A2	0.125	0.195	3.18	4.95	-
B	0.014	0.022	0.356	0.558	-
B1	0.030	0.070	0.77	1.77	8
C	0.008	0.015	0.204	0.381	-
D	1.980	2.095	50.3	53.2	5
D1	0.005	-	0.13	-	5
E	0.600	0.625	15.24	15.87	6
E1	0.485	0.580	12.32	14.73	5
e	0.100 BSC		2.54 BSC		-
e_A	0.600 BSC		15.24 BSC		6
e_B	-	0.700	-	17.78	7
L	0.115	0.200	2.93	5.08	4
N	40		40		9

Rev. 0 12/93

NOTES:

1. Controlling Dimensions: INCH. In case of conflict between English and Metric dimensions, the inch dimensions control.
2. Dimensioning and tolerancing per ANSI Y14.5M-1982.
3. Symbols are defined in the "MO Series Symbol List" in Section 2.2 of Publication No. 95.
4. Dimensions A, A1 and L are measured with the package seated in JEDEC seating plane gauge GS-3.
5. D, D1, and E1 dimensions do not include mold flash or protrusions. Mold flash or protrusions shall not exceed 0.010 inch (0.25mm).
6. E and e_A are measured with the leads constrained to be perpendicular to datum -C-.
7. e_B and e_C are measured at the lead tips with the leads unconstrained. e_C must be zero or greater.
8. B1 maximum dimensions do not include dambar protrusions. Dambar protrusions shall not exceed 0.010 inch (0.25mm).
9. N is the maximum number of terminal positions.
10. Corner leads (1, N, N/2 and N/2 + 1) for E8.3, E16.3, E18.3, E28.3, E42.6 will have a B1 dimension of 0.030 - 0.045 inch (0.76 - 1.14mm).

Plastic Leaded Chip Carrier Packages (PLCC)



N44.65 (JEDEC MS-018AC ISSUE A)
44 LEAD PLASTIC LEADED CHIP CARRIER PACKAGE

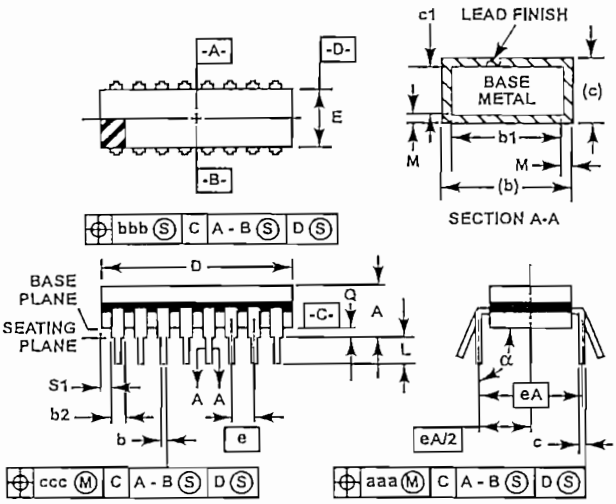
SYMBOL	INCHES		MILLIMETERS		NOTES
	MIN	MAX	MIN	MAX	
A	0.165	0.180	4.20	4.57	-
A1	0.090	0.120	2.29	3.04	-
D	0.685	0.695	17.40	17.65	-
D1	0.650	0.656	16.51	16.66	3
D2	0.291	0.319	7.40	8.10	4, 5
E	0.685	0.695	17.40	17.65	-
E1	0.650	0.656	16.51	16.66	3
E2	0.291	0.319	7.40	8.10	4, 5
N	44		44		6

Rev. 2 11/97

NOTES:

1. Controlling dimension: INCH. Converted millimeter dimensions are not necessarily exact.
2. Dimensions and tolerancing per ANSI Y14.5M-1982.
3. Dimensions D1 and E1 do not include mold protrusions. Allowable mold protrusion is 0.010 inch (0.25mm) per side. Dimensions D1 and E1 include mold mismatch and are measured at the extreme material condition at the body parting line.
4. To be measured at seating plane -C- contact point.
5. Centerline to be determined where center leads exit plastic body.
6. "N" is the number of terminal positions.

Ceramic Dual-In-Line Frit Seal Packages (CERDIP)



F40.6 MIL-STD-1835 GDIP1-T40 (D-5, CONFIGURATION A)
40 LEAD CERAMIC DUAL-IN-LINE FRIT SEAL PACKAGE

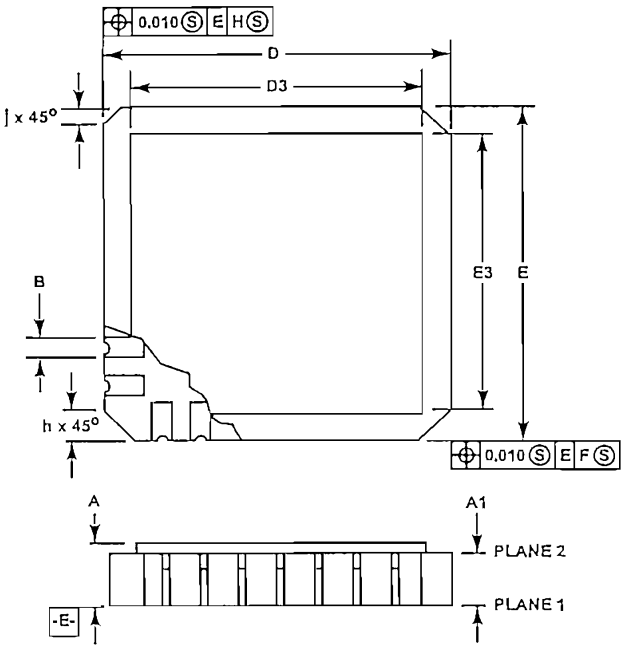
SYMBOL	INCHES		MILLIMETERS		NOTES
	MIN	MAX	MIN	MAX	
A	-	0.225	-	5.72	-
b	0.014	0.026	0.36	0.66	2
b1	0.014	0.023	0.36	0.58	3
b2	0.045	0.065	1.14	1.65	-
b3	0.023	0.045	0.58	1.14	4
c	0.008	0.018	0.20	0.46	2
c1	0.008	0.015	0.20	0.38	3
D	-	2.096	-	53.24	5
E	0.510	0.620	12.95	15.75	5
e	0.100 BSC		2.54 BSC		-
eA	0.600 BSC		15.24 BSC		-
eA/2	0.300 BSC		7.62 BSC		-
L	0.125	0.200	3.18	5.08	-
Q	0.015	0.070	0.38	1.78	6
S1	0.005	-	0.13	-	7
α	90°	105°	90°	105°	-
aaa	-	0.015	-	0.38	-
bbb	-	0.030	-	0.76	-
ccc	-	0.010	-	0.25	-
M	-	0.0015	-	0.038	2, 3
N	40		40		8

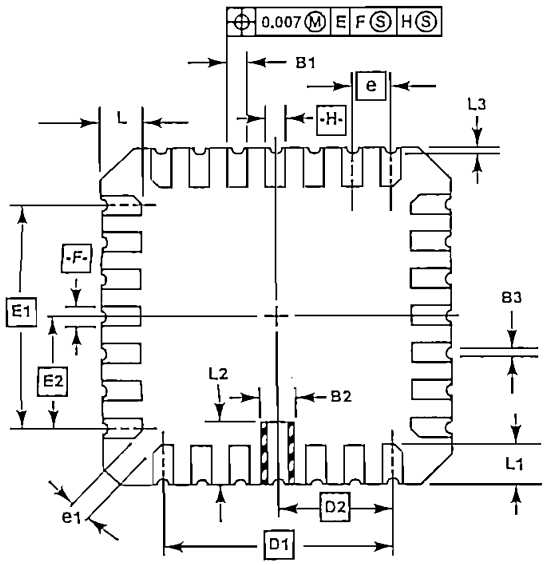
Rev. 0 4/94

NOTES:

1. Index area: A notch or a pin one identification mark shall be located adjacent to pin one and shall be located within the shaded area shown. The manufacturer's identification shall not be used as a pin one identification mark.
2. The maximum limits of lead dimensions b and c or M shall be measured at the centroid of the finished lead surfaces, when solder dip or tin plate lead finish is applied.
3. Dimensions b1 and c1 apply to lead base metal only. Dimension M applies to lead plating and finish thickness.
4. Corner leads (1, N, N/2, and N/2+1) may be configured with a partial lead paddle. For this configuration dimension b3 replaces dimension b2.
5. This dimension allows for off-center lid, meniscus, and glass overrun.
6. Dimension Q shall be measured from the seating plane to the base plane.
7. Measure dimension S1 at all four corners.
8. N is the maximum number of terminal positions.
9. Dimensioning and tolerancing per ANSI Y14.5M - 1982.
10. Controlling dimension: INCH.

Ceramic Leadless Chip Carrier Packages (CLCC)





J44.A MIL-STD-1835 CQCC1-N44 (C-5)
44 PAD CERAMIC LEADLESS CHIP CARRIER PACKAGE

SYMBOL	INCHES		MILLIMETERS		NOTES
	MIN	MAX	MIN	MAX	
A	0.064	0.120	1.63	3.05	6, 7
A1	0.054	0.088	1.37	2.24	-
B	0.033	0.039	0.84	0.99	4
B1	0.022	0.028	0.56	0.71	2, 4
B2	0.072 REF		1.83 REF		-
B3	0.006	0.022	0.15	0.56	-
D	0.640	0.662	16.26	16.81	-
D1	0.500 BSC		12.70 BSC		-
D2	0.250 BSC		6.35 BSC		-
D3	-	0.662	-	16.81	2
E	0.640	0.662	16.26	16.81	-
E1	0.500 BSC		12.70 BSC		-
E2	0.250 BSC		6.35 BSC		-
E3	-	0.662	-	16.81	2
e	0.050 BSC		1.27 BSC		-
e1	0.015	-	0.38	-	2
h	0.040 REF		1.02 REF		5
j	0.020 REF		0.51 REF		5
L	0.045	0.055	1.14	1.40	-
L1	0.045	0.055	1.14	1.40	-
L2	0.075	0.095	1.90	2.41	-
L3	0.003	0.015	0.08	0.38	-
ND	11		11		3
NE	11		11		3
N	44		44		3

Rev. 0 5/18/94

NOTES:

1. Metallized castellations shall be connected to plane 1 terminals and extend toward plane 2 across at least two layers of ceramic or completely across all of the ceramic layers to make electrical connection with the optional plane 2 terminals.
2. Unless otherwise specified, a minimum clearance of 0.015 inch (0.38mm) shall be maintained between all metallized features (e.g., lid, castellations, terminals, thermal pads, etc.)
3. Symbol "N" is the maximum number of terminals. Symbols "ND" and "NE" are the number of terminals along the sides of length "D" and "E", respectively.
4. The required plane 1 terminals and optional plane 2 terminals (if used) shall be electrically connected.
5. The corner shape (square, notch, radius, etc.) may vary at the manufacturer's option, from that shown on the drawing.
6. Chip carriers shall be constructed of a minimum of two ceramic layers.
7. Dimension "A" controls the overall package thickness. The maximum "A" dimension is package height before being solder dipped.
8. Dimensioning and tolerancing per ANSI Y14.5M-1982.
9. Controlling dimension: INCH.