

**UNIVERSIDAD DON BOSCO
FACULTAD DE INGENIERIA**

TEMA :

**“SISTEMA DE ADQUISICION DE DATOS DE PRESION,
TEMPERATURA Y NIVEL PARA EL CONTROL
AUTOMATICO DE PROCESOS MANEJADO POR
COMPUTADORA “**

**TRABAJO DE GRADUACION PREPARADO
PARA LA FACULTAD DE INGENIERIA**



**PARA OPTAR AL GRADO DE
INGENIERO EN ELECTRONICA**

PRESENTADO POR

**WILLIAM EDGARDO ACEVEDO MORAN
JUAN CARLOS BLANCO UMAÑA
GERMAN ENRIQUE RODRIGUEZ PEREIRA**

SOYAPANGO


EL SALVADOR

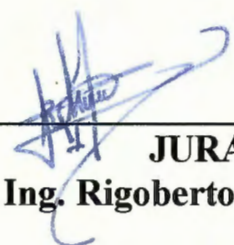
CENTRO AMERICA

UNIVERSIDAD DON BOSCO
FACULTAD DE INGENIERIA

TEMA :

**“SISTEMA DE ADQUISICION DE DATOS DE PRESION,
TEMPERATURA Y NIVEL PARA EL CONTROL
AUTOMATICO DE PROCESOS MANEJADO POR
COMPUTADORA “**

(F) 
ASESOR
Ing. Rafael Mayorga

(f) 

JURADO
Ing. Rigoberto Chinchilla

(f) _____
JURADO
Ing. Giovanni Vizcarra

AGRADECIMIENTOS

Quiero agradecer a Dios todo poderoso por permitirme llegar hasta aquí, a La Virgen María que siempre ha estado conmigo, “Realmente me siento bendecido”. Agradezco especialmente a mis padres, María del Carmen Umaña, Fidel Angel Blanco que me han brindado su apoyo, comprensión y han estado en el momento que los he necesitado “Aquí esta su tesis, Gracias por todo”.

A mis hermanos Fidel, Ricardo, Elena, Ricardo, Hector y Margarita. “El triunfo es de ustedes”.

A Beth por su comprensión.

Y muy en especial a todos los amigos que me han brindado soporte, asistencia y mucha ayuda si se me queda alguien discúlpenme: William, Enrique, Ernesto, Franco.

“KEEP THE FAITH”

Juan Carlos

Deseo agradecer al Señor Todo Poderoso por haberme dado la oportunidad de alcanzar este triunfo. A mis padres por apoyarme e inculcarme el amor a Dios, al estudio y al trabajo. A mis hermanos, tíos, primos y amigos por su comprensión.

Agradezco a mis profesores, compañeros de estudio, y especialmente a mis compañeros de tesis: William y Juan Carlos, por su comprensión y confianza.

Gracias a todos, les dedico este triunfo!

German Enrique Rodríguez.

Agradezco Eternamente a Dios Todo Poderoso, a mis Padres, a mi Familia y a todos mis amigos, tanto en mi trabajo como en mi vida personal.

William Edgardo Acevedo Morán

INDICE GENERAL

INTRODUCCION.....	i
OBJETIVOS	iii

CAPITULO 1

1. SISTEMAS DE ADQUISICION DE DATOS.....	1
1.1 SISTEMAS TIPICOS	1
1.1.1 Multiplexor análogo	1
1.1.2 Acondicionador de señal	2
1.1.3 Convertidor de análogo a digital.....	2
1.1.4 Controlador E/S.....	2
1.1.5 Computadora.....	2
1.1.6 Buffer de salida.....	2
1.1.7 Periféricos	3
1.2 USOS BASICOS DE UN SISTEMA DE ADQUISICION DE DATOS.....	3
1.2.1 Introducción de datos.....	3
1.2.2 Análisis de señal.....	4
1.2.3 Control de procesos	4
1.3 CONCEPTOS DE MUESTREO	4
1.3.1 Muestreo	5
1.3.2 Cuantización	5
1.3.3 Procesamiento digital	5
1.3.4 Recuperación	5
1.4 PARAMETROS DEL SISTEMA	6
1.4.1 Exactitud relativa	6
1.4.2 Precisión.....	6
1.4.3 Resolución.....	6



1.4.4 Sobrerango.....	7
1.4.5 Monotonía.....	7
1.4.6 Coeficientes de temperatura	7
1.4.7 Velocidad de conversión.....	7
1.4.8 Linealidad.....	7

CAPITULO 2

2. TRANSDUCTORES.....	9
2.1 TRANSDUCTORES DE PRESION	9
2.1.1 Transductores de presión de deformación elástica.....	10
2.1.1.1 <i>El tubo de Bourdon</i>	10
2.1.1.2 <i>Transductor de presión en espiral</i>	11
2.1.1.3 <i>Transductor de presión helicoidal</i>	12
2.1.1.4 <i>Transductor de presión en fuelle</i>	13
2.1.1.5 <i>Transductor de presión de diafragma</i>	13
2.1.2 Transductores de presión piezo-eléctricos.....	14
2.1.3 Transductor de presión en circuitos integrados.....	14
2.2 TRANSDUCTORES DE NIVEL.....	15
2.2.1 Transductor de nivel capacitivo.....	15
2.2.3 Sensor de nivel de electrodos	16
2.3 SENSORES DE TEMPERATURA.....	17
2.3.1 Sensores de temperatura semiconductores.....	17
2.3.2 Termocuplas	18

CAPITULO 3

3. CONVERTIDORES ANALOGO A DIGITAL Y DIGITAL A ANALOGO	19
3.1 METODOS DE CONVERSION DE A/D.....	19

3.2 CRITERIOS DE SELECCION PARA CONVERTIDORES A/D.....	20
3.2.1 Número de bits	20
3.2.2 Velocidad de conversión	21
3.2.3 Exactitud de la conversión	21
3.3 TECNICAS DE CONVERSION	21
3.3.1 Convertidores de aproximación sucesiva.....	22
3.3.1.1 Asignación de pesos por bit de un número binario	24
3.3.1.2 Creando un número digital	24
3.4 TIPOS DE DISPOSITIVOS	25
3.4.1 ADC0809 convertidor CMOS análogo a digital con multiplexor de 8 canales	25
3.4.1.2 Descripción.....	26
3.5 CONVERTIDOR DIGITAL ANALOGO	27

CAPITULO 4

4. DISPOSITIVOS DE INTERFASE.....	30
4.1 MANEJADORES DE LINEA	30
4.2 EL SN74LS244 / SN74LS241	30
4.3 DECODIFICADORES/DEMULTIPLEXORES	31
4.3.1 El decodificador/demultiplexor 74LS138	31
4.4 EL PPI 8255 A.....	33
4.4.1 Bus bidireccional de datos.....	33
4.4.2 Lógica de control de lectura/escritura	33
4.4.3 Operación básica	34
4.4.4 Características de cada puerto.....	34
4.4.5 Modos de operación e inicialización.....	35
4.4.5.1 Modo 0	36
4.4.5.2 Modo 1	37
4.4.5.3 Modo 2	38

4.5 DIRECCIONAMIENTO DE LA COMPUTADORA HACIA EL PPI8255A	38
---	-----------

CAPITULO 5

5.0 ETAPA DE CONTROL	41
5.1 SISTEMA DE CONTROL DE LAZO CERRADO.....	41
5.2 SISTEMA DE CONTROL DE LAZO ABIERTO	42
5.3 ACCIONES BASICAS DE CONTROL Y CONTROLADORES	42
5.3.1 Acciones Básicas de control.....	42
5.4 CRITERIOS DE EVALUACION DE UN SISTEMA DE CONTROL	45
5.5 ANALISIS Y DISEÑO DE LOS SISTEMAS DE CONTROL.....	46
5.6 PROBLEMAS POTENCIALES EN LOS SISTEMAS DE CONTROL.....	47

CAPITULO 6

6. DESCRIPCION DEL HARDWARE DEL SISTEMA.....	49
6.1 ETAPA DE ADQUISICIÓN	49
6.1.1 Circuito de adquisición.....	50
6.2 ETAPA DE CONTROL	55
6.2.1 Control de Temperatura	55
6.2.2 Control de presión.....	56
6.2.3 Control de nivel.....	56

CAPITULO 7

7. SOFTWARE.....	58
7.1 SISTEMA OPERATIVO WINDOWS.....	58
7.2 VENTAJAS DEL SISTEMA OPERATIVO WINDOWS	59
7.2.1 Multitarea	60

7.2.2 Objetos Manejados por eventos	60
7.2.3 El Clipboard o Portapapeles	60
7.2.4 Uso del OLE	60
7.2.5 DDE	61
7.2.6 Configuración de recursos para Windows en general	61
7.3 LENGUAJE DE PROGRAMACION VISUAL BASIC	61
7.4 CARACTERISTICAS DE VISUAL BASIC	62
7.5 REQUERIMIENTOS DE HARDWARE PARA EJECUTAR VISUAL BASIC ...	62
7.6 EL AMBIENTE DE VISUAL BASIC	63
7.6.1 La pantalla principal (Main Screen)	63
7.6.2 Titulo (title bar)	64
7.6.3 Barra de menús (menús bar)	64
7.7 LOS ELEMENTOS DE VISUAL BASIC	65
7.7.1 La ventana de proyectos (project window)	66
7.7.2 Los forms	66
7.7.3 Los objetos	66
7.7.4 Ventana de Propiedades	66
7.7.5 Los eventos	66
7.7.6 Las Subrutinas	67
7.7.7 Los módulos	67
7.7.8 La ventana de códigos	67
7.7.9 La caja de herramientas	67
7.8 COMO DESARROLLAR UNA APLICACION EN VISUAL BASIC	68
7.9 COMANDOS DE VISUAL BASIC	70
7.10 FUNCIONAMIENTO DEL PROGRAMA DE CONTROL	74
7.10.1 Diagramas de flujo	74
7.11 EXPLICACION DEL PROGRAMA	74
7.11.1 Opción ejecutar	84
7.11.2 Opción salir	88
7.11.3 Opción consulta	88

7.11.4 Subrutina bin	89
7.12 PROGRAMA PRINCIPAL	89
7.12.1 Presenta	90
7.12.2 Inivar	92
7.12.3 Fijarval	93
7.12.4 Lectura	98
7.12.5 About	106
CONCLUSIONES	107
GLOSARIO TECNICO	108
ANEXO 1 Sensor SPX100	110
ANEXO 2 Termocupla tipo J	113
ANEXO 3 Diagrama de Bloques del ADC0809	114
ANEXO 4 Diagrama de Tiempo del ADC0809	115
ANEXO 5 Buffers de la Familia SN74'	116
ANEXO 6 Decodif. / Demul. 74LS138	118
ANEXO 7 Modo 0 de la PPI	120
ANEXO 8 Diagrama del Circuito	121
ANEXO 9 Amplificadores Operacionales	122
BIBLIOGRAFIA	124

INDICE DE FIGURAS

Figura 1. Sistema de Adquisición de Datos Típico	3
Figura 2a. No linealidad Integral.....	8
Figura 2b. No linealidad diferencial.....	8
Figura 3. Tubo de Bourdon	11

Figura 4. Transductor de presión en espiral.....	12
Figura 5. Transductor de presión helicoidal.....	12
Figura 6. Transductor de Presión de Diafragma Integrado.....	15
Figura 7. Transductor de nivel capacitivo.....	16
Figura 8. Diagrama a bloques de un convertidor de aproximación sucesiva	23
Figura 9. Formación de un numero digital.....	26
Figura 10 Convertidor R-2R.....	29
Figura 11. Diagrama a bloques del PPI	36
Figura 12. Ensamble de la palabra de control	39
Figura 13a Controlador on - off.....	44
Figura 13b Controlador on - off con brecha diferencial o zona muerta.....	44
Figura 14 Nivel de h (t) en función de t.....	45
Figura 15 Diagrama esquemático de un controlador.....	47
Figura 16. Detector de nivel	54
Figura 17 Diagrama del circuito manejador de relé.....	57
Figura 18. Ambiente Windows.....	59
Figura 19. pantalla principal de Visual Basic	65
Figura 20. Subrutina presenta y Load.....	75
Figura 21. Subrutina de adquisición	81
Figura 22. Subrutina DEC to BIN y de retardo	82
Figura 23. Pantalla de presentación.....	84
Figura 24. Form de fijarval	85
Figura 25. Pantalla de adquisición de datos.....	88

INDICE DE TABLAS

Tabla 1. Comparación de las características del ADC.....	20
Tabla 2. Tabla de códigos de salida del 74LS138	31

Tabla 3. Tabla de conexión del 74LS138 y el slot	32
Tabla 4. Tabla de selección de registro del PPI.	34
Tabla 5. Operación básica del PPI.....	35
Tabla 6. Direccionamiento de las variables.....	51
Tabla 7. Direccionamiento para registro del PPI.	52

INTRODUCCION

El proyecto consiste en proveer un sistema de adquisición de datos y sistemas de control, el cual incluye todos los componentes de hardware y software necesarios para su funcionamiento. El sistema incluye una computadora PC AT compatible, junto con todos los transductores y sensores necesarios para aplicaciones individuales tales como: Control de temperatura en hornos, control de presión en sistemas fluidicos, control de medición y detección de nivel en líquidos.

El funcionamiento del diseño a implementado es el siguiente: se sensan las variables a controlar, luego estas señales son transformadas de analógicas a digitales, las señales digitales son introducidos al computador por medio de uno de sus slots, los datos son leídos por un manejador de bases de datos, en el cual el usuario fija los valores necesarios del proceso a controlar; de la base de datos se obtienen gráficos para tener una visualización de los resultados, seguidamente la salida del control del proceso se obtiene de un puerto de salida del computador para ser introducida a una interfase de potencia, la cual maneja los actuadores que se encargan de corregir las variables que se sensan, según los valores fijados por el usuario a través del programa, obteniéndose de esta manera el control automático del proceso.

El presente documento esta dividido en siete capítulos, en cada uno de ellos se profundiza sobre diferentes aspectos o áreas que componen dicho trabajo.

Inicialmente en el capitulo uno se detalla sobre los sistemas de adquisición de datos, bajo un enfoque teórico, exponiendo y analizando criterios que deben considerarse en los diseños de sistemas de este tipo.

Seguidamente en el capitulo dos se discute el tema de transductores, donde se analizan y se explican sus aplicaciones y características para cada uno de ellos.

A continuación el capitulo tres trata sobre los DAC y ADC, en el cual se hace una comparación entre todos los tipos de ADC que existen, se explica sobre su funcionamiento.

En el capítulo cuatro se explican los dispositivos de interfase que se emplearon para lograr adquirir y controlar las variables, tales como la PPI, buffers y amplificadores operacionales entre otros.

En el mismo orden el siguiente capítulo trata sobre la etapa de control, en el cual se analizan los diferentes tipos de sistemas de control.

Finalmente los dos últimos capítulos comprenden el funcionamiento del hardware y del software del sistema respectivamente.

OBJETIVOS

- Proveer un sistema de control automático, del tipo encendido apagado y proporcional, de bajo costo, controlado por computadora bajo ambiente Windows.
- Facilitar el análisis de variables, por medio de comparaciones, análisis estadísticos, gráficos y registros de información que definitivamente facilitan la toma de decisiones para optimizar el control de calidad de una planta.
- Poner en práctica los conocimientos teóricos y prácticos adquiridos para optar al título de Ingeniero en Electrónica.
- Proporcionar un proyecto el cual hará un uso más eficiente de la materia prima del proceso en el que intervenga, reduciendo los desperdicios que se generan por mal funcionamiento del sistema que se emplea.
- Proveer un sistema controlado bajo software por medio de interfaces visuales fáciles de interpretar.
- Sentar las bases para investigaciones y desarrollo futuros en sistemas de control automático.
- Construir un sistema con elementos y dispositivos disponibles en el Mercado Nacional.

1. SISTEMAS DE ADQUISICION DE DATOS

Un sistema de Adquisición de datos en el amplio sentido podría significar cualquier método para adquirir cualquier tipo de datos, sin embargo un sistema de adquisición de datos (DAS) usualmente denota un grupo de dispositivos electrónicos que permiten la medición, cuantización y conversión de señales eléctricas análogas a un formato digital que puede ser leído por la computadora y de esta manera la información en forma digital se puede procesar, almacenar, comunicar y presentar con facilidad, en forma indestructible y sin errores. Las señales pueden ser producidas por sensores de temperatura, transductores de presión, medidores de flujo, etc. Con un sistema de adquisición, la computadora puede ser utilizada para recolectar, monitorear, presentar y analizar los datos. Si el sistema de adquisición tiene capacidades de salida, la computadora puede ser utilizada para controlar con exactitud los procesos y lograr una eficiencia máxima.

1.1 SISTEMAS TIPICOS.

Un voltímetro digital es un DAS que simplemente presenta su salida como un valor numérico. Un proceso de control automático con un DAS manejado por una computadora provee un mayor rango de funciones de adquisición de datos. Un sistema típico de adquisición de datos con una computadora se muestra en la figura 1. Las partes claves del sistema incluyen:

1.1.1 Multiplexor análogo

Esta función permite la selección de cualquiera de las señales análogas de entrada, a través de un registro de direcciones el cual se encarga de proveer el código para el direccio-

namiento de cada canal de entrada.

1.1.2 Acondicionador de señal

Esta función usualmente sigue al multiplexor, pero podría ser utilizada en todas las líneas de entrada de señal para conversión. Los requisitos de acondicionamiento podrían ser cualquiera de los siguientes: Amplificación lineal o logarítmica, Filtrado, Detección de picos, muestra y retención.

1.1.3 Convertidor de análogo a digital

Esta función permite que el mundo analógico se comunique con los circuitos digitales (computadores) que se encargan de llevar a cabo las tareas de almacenamiento, manipulación, comparación y cálculo.

1.1.4 Controlador E/S

El controlador de entrada-salida, genera la temporización del sistema y controla las funciones de read/write de la memoria, para que los datos puedan ser introducidos, procesados, y de esta forma obtener una salida.

1.1.5 Computadora

Un sistema operacional de computadora basado en un microprocesador. Que posee una unidad de almacenamiento y es controlada por software. Usualmente se vale de periféricos para interactuar con el mundo exterior.

1.1.6 Buffer de salida

El buffer de salida combina el dato con el formato propio de la señal para la salida a un dispositivo periférico específico. El buffer de salida es llamado también controlador de periféricos.

1.1.7 Periféricos

Diferentes dispositivos de entrada o salida de un computador, los cuales sirven para comunicar al computador con el mundo exterior de esta; monitor, teclado, mouse, impresor son ejemplos de periféricos comúnmente usados.

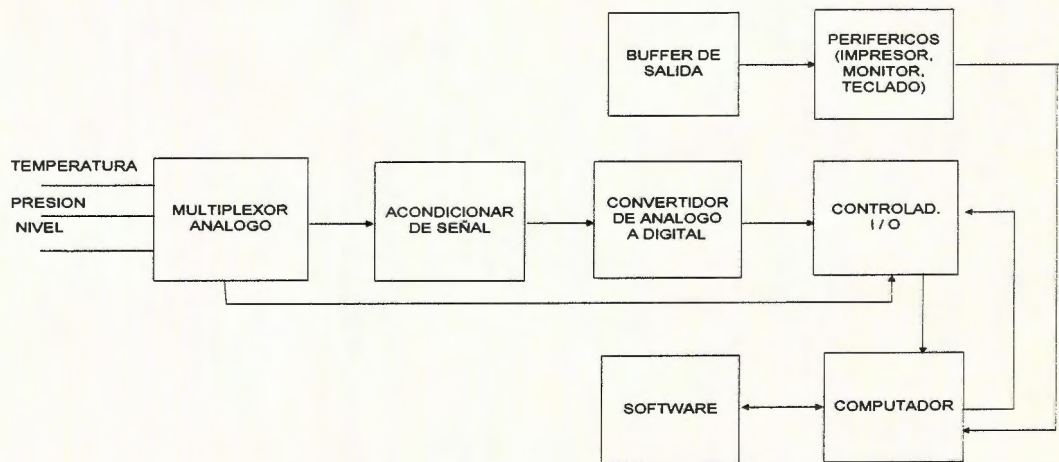


Figura 1. Sistema de Adquisición de Datos Típico

1.2 USOS BASICOS DE UN SISTEMA DE ADQUISICION DE DATOS

Cualquier sistema de hardware que es análogo y digital tendrá un sistema de adquisición de datos incluido. El sistema puede ser utilizado para :

1. Introducción de datos
2. Análisis de señal
3. Control de Procesos

1.2.1 Introducción de datos

La introducción de datos incluye no solamente la medición de los datos análogos sino que incluye el traslado de resultados a señales digitales y el almacenamiento de datos

para el análisis y proceso futuro. Estas funciones pueden ser desarrolladas en intervalos precisos de tiempo y con exactitudes que van más allá de las posibilidades humanas.

1.2.2 Análisis de señal

Muchos procesos generan señales que varían con el tiempo de las cuales se deben de extraer datos. Como ejemplos tenemos: control de tráfico aéreo, exploración sísmica, y monitoreo de pacientes en los hospitales. El método más común de analizar esta información es trasladarla en una serie de Fourier apropiada para analizarla en el dominio de la frecuencia.

1.2.3 Control de procesos

Por muchos años los sistemas de control de procesos eran controlados manualmente. Actualmente se logra por medio de DAS, con los cuales se monitorean cientos de entradas, se presentan las lecturas, se hacen cálculos y se da una retroalimentación instantánea al usuario. El operador toma una acción apropiada basado en la salida del computador. Mientras más complejos se vuelven los sistemas, más decisiones son tomadas por el computador.

1.3 CONCEPTOS DE MUESTREO

Para que un sistema de adquisición de datos funcione, la señal de entrada, debe ser muestreada. El muestreo puede ser definido como la medición de una señal continua en intervalos discretos de tiempo. Estas señales muestreadas representan algún parámetro análogo convertido en una serie de valores discretos.

Un sistema de adquisición de datos esta constituido por los procesos descritos a continuación.

1.3.1 Muestreo

Muestrear una señal significa conservar solamente los valores instantáneos tomados a intervalos regulares de tiempo, la frecuencia de estos intervalos viene determinada por el teorema de muestreo, conocido usualmente como el criterio de Nyquist el cual establece que las señales con un ancho de banda finito de frecuencia f -Hertz pueden ser descritas si son muestreadas a instantes de tiempo separados por $T = (1/2) f$ segundos. Básicamente, esto significa que las señales deben ser muestreadas a una frecuencia al menos dos veces mayor que la frecuencia de la señal que esta siendo analizada. En la práctica se utiliza una frecuencia de muestreo mayor de la establecida por el criterio de Nyquist, con el fin de evitar problemas con ciertas formas de onda.

1.3.2 Cuantización

Para que una computadora u otro equipo digital procese señales análogas, estas deben ser convertidas de el dominio análogo al dominio digital. Este proceso es llamado cuantización. La cuantización podría ser definida como la conversión de una función de entrada la cual puede tener valores en un rango continuo a una salida que tiene únicamente valores discretos.

1.3.3 Procesamiento digital

Evaluación de la función lineal como una serie de valores discretos, que son comparados, analizados para poder tomar una decisión en base a los datos suministrados por las diferentes fuentes de información análoga.

1.3.4 Recuperación

Por medio de un convertidor de digital a analógico que retorna la información procesada en una forma análoga para poder controlar algún tipo de actuador que se encarga de llevar acabo una acción en base a la discrepancia de los datos que están siendo medidos.

1.4 PARAMETROS DEL SISTEMA

Para diseñar un sistema de adquisición de datos, los parámetros del sistema deben ser definidos. Diferentes modelos de dispositivos de adquisición de datos se encuentran disponibles en el mercado. Los diseñadores especifican los parámetros en diferentes formas. Sin embargo existen factores y parámetros básicos claves para la selección de un sistema de adquisición de datos en una aplicación particular. Los parámetros de mayor importancia a considerar son los siguientes:

1.4.1 Exactitud relativa

Esta es una medida de la habilidad del sistema de adquisición de producir códigos de salida correctos, para todos los voltajes de entrada posibles, en relación a su rango de escala completa. La exactitud relativa es una función directa de la linealidad del sistema.

La forma típica de establecer la exactitud es:

$\pm 0.01\%$ del valor FSR (donde: FSR = Lectura de escala completa)

$\pm 1/2$ LSB (error de cuantización donde: LSB = Bit menos significativo)

1.4.2 Precisión

La precisión es una medida de la redundancia del sistema. En otras palabras, es la habilidad que tiene el sistema para producir el mismo código de salida cuando se hacen mediciones sucesivas del mismo valor de entrada. El ruido generado internamente por el sistema es una causa de una precisión pobre.

1.4.3 Resolución

Es la habilidad de un sistema de adquisición de datos para distinguir entre niveles de entrada análogos adyacentes. Un sistema de 8 bits podría ser capaz de distinguir entre niveles de entrada que difieren por $1/256$ del rango de escala completa. La resolución

también define el rango dinámico del sistema el cual puede ser establecido de la siguiente forma:

$$256 \text{ a } 1 \text{ o } (\text{dB} = 20 \log_{10} 256) = 48 \text{ dB}$$

1.4.4 Sobrerango

Cuando el valor normal de voltaje de escala completa es excedido, la salida continua registrando el código asignado al valor de escala completa.

1.4.5 Monotonía

La monotonía en un sistema de adquisición de datos, es una garantía de que el código de salida del sistema continuará incrementando (o decrementando) con una entrada continuamente incrementando (o decrementando). Este problema es más probable que ocurra en los puntos medios o cuartos de la lectura de escala completa.

1.4.6 Coeficientes de temperatura

Los cambios en la temperatura ambiente afectan la habilidad del sistema para producir mediciones exactas. Los problemas mayores con temperatura son los cambios de ganancia y el desplazamiento del punto cero. Los coeficientes de temperatura son factores que no se deben dejar de considerar.

1.4.7 Velocidad de conversión

La velocidad de conversión provee una indicación de la frecuencia de muestreo y determina la cantidad de conversiones que el sistema realizará por unidad o fracción de tiempo.

1.4.8 Linealidad

Una de las principales desventajas que poseen la mayoría de transductores es que su

salida no es lineal, esto genera un error de linealidad, este consiste en la desviación de la curva de salida del dispositivo con respecto a una línea recta en el plano de la función de transferencia.

Se pueden diferenciar los errores de transferencia en dos tipos principales, el primero es llamado no linealidad integral, el cual es la máxima desviación de una recta perfecta dibujada en el plano de la función de transferencia, Ver figura 2a, esta es especificada usualmente como porcentaje de la escala total, o en quantum (tal como $\pm 1/2$ LSB). El segundo tipo de error es llamado no linealidad diferencial, esta es la medida de la máxima desviación de cualquier intervalo de quantum en la función de transferencia del sistema del valor teórico, esta puede ser especificada como un porcentaje del nivel de 1 quantum o LSB. Ver figura 2b.

En la práctica se pueden alcanzar a través de métodos muy sofisticados, aproximaciones muy buenas a una curva lineal de transferencia a lo largo de cualquier coordenada del plano cartesiano de función de transferencia, ya que la linealidad es vulnerable a factores externos e internos al sistema, tal como temperatura, frecuencia, etc.

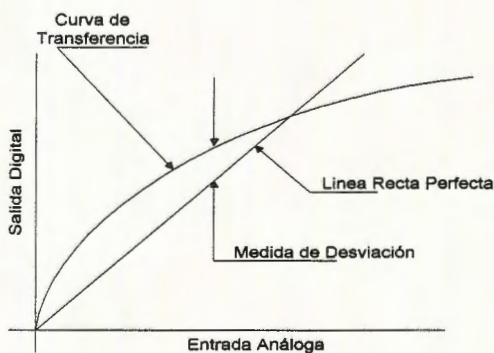


Figura 2a. No linealidad Integral

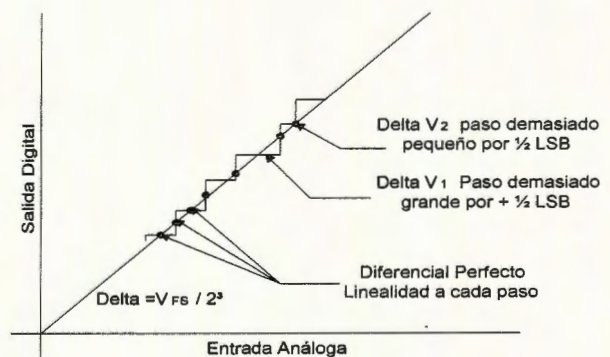


Figura 2b. No linealidad diferencial

2. TRANSDUCTORES

En un sistema de control de procesos industriales, cada elemento realiza una función específica, así, el elemento primario conocido como transductor, se encarga de la medición de la variable controlada. El transductor convierte el valor de la variable en proceso en una señal de magnitud proporcional que sea compatible con el siguiente elemento del sistema de control, el cual puede ser un transmisor, un controlador, un registrador o un indicador.

La función del transmisor, es acondicionar la señal proveniente del transductor, en valores normalizados de corriente o voltaje para ser llevados al controlador. El controlador es el cerebro del lazo de control, vigila el proceso y decide las acciones a tomar para corregir los errores que se presentan en la variable controlada.

Otro elemento importante en un sistema de control de procesos es el elemento final de control, conocido también como actuador. Con este elemento se ejerce la acción de control directamente sobre la variable a controlar, que para el presente trabajo serán nivel, temperatura y presión. A continuación se describen los tipos de transductores más usados en la industria.

2.1 TRANSDUCTORES DE PRESION

La medición de la presión es importante puesto que es una variable encontrada con mucha frecuencia en la industria ya sea en ventiladores, compresores, calderas, bombas, etc. además se puede inferir el valor de otras variables como son caudal, nivel, etc.

La presión es una fuerza por unidad de superficie y puede expresarse en unidades tales como kg/cm^2 , psi, bar y atmósferas; una medida de presión que se ha tratado de estandarizar es el Pascal (N/m^2).

La presión puede medirse en valores absolutos o diferenciales. La presión absoluta (P_a), se mide con relación al cero absoluto, la presión atmosférica (P_{at}) es la presión ejercida por la atmósfera terrestre medida mediante un barómetro. La presión relativa o manométrica (P_g), es la determinada por un elemento que mide la diferencia entre la presión absoluta y la atmosférica del lugar donde se efectúa la medición. La presión diferencial es la diferencia de dos presiones entre dos puntos cualesquiera.

El vacío es la diferencia de presiones entre la presión atmosférica, y la presión absoluta. El campo de aplicación de los transductores es muy amplio y abarca desde valores muy pequeños hasta presiones de miles de kg/cm^2 .

2.1.1 Transductores de presión de deformación elástica

Este tipo de transductor convierte el valor de la presión en un desplazamiento proporcional, aprovechando el comportamiento elástico que presentan algunos materiales cuando se someten a algún esfuerzo, sin sobrepasar el punto de fluencia del material, esto significa que el dispositivo volverá a su forma original una vez desaparece el esfuerzo que produjo la deformación. Dentro de este tipo de transductores se tienen:

2.1.1.1 El tubo de Bourdon

El elemento de Bourdon, es un tubo de sección elíptica que forma un anillo casi completo, cerrado por un extremo. Al aumentar la presión en el interior del tubo, este tiende a enderezarse y el movimiento es transmitido a la aguja indicadora, por un sector dentado y un piñón. La ley de deformación del tubo de Bourdon es muy compleja y ha sido determinada empíricamente. El material empleado en el tubo de Bourdon normalmente es acero inoxidable.

El tubo de Bourdon mide presiones manométricas y de vacío, y puede utilizarse para medir presiones en rangos comprendidos desde 0 hasta 10,000 psig, según el material. En la figura 3 se muestran las partes de un tubo de Bourdon.

2.1.1.2 Transductor de presión en espiral

Estos transductores se construyen en base a tubos de Bourdon, por lo tanto, su principio de funcionamiento es el mismo, pueden utilizarse para medir presiones manométricas y de vacío. Consta básicamente, de varias vueltas enrolladas en espiral alrededor de un eje común, cuando se somete a presión tiende a desenrollarse proporcionando un desplazamiento. Este tipo de transductor está restringido a la medida de presiones intermedias, aproximadamente hasta 300 psig, porque a presiones muy altas presenta el inconveniente de que sus paredes tratan de unirse. El elemento en espiral se construye de diferentes materiales y aleaciones (bronce, acero inoxidable, berilio-cobre, etc.) de acuerdo al rango de presión a medir. En la figura 4 se muestra un transductor en espiral.

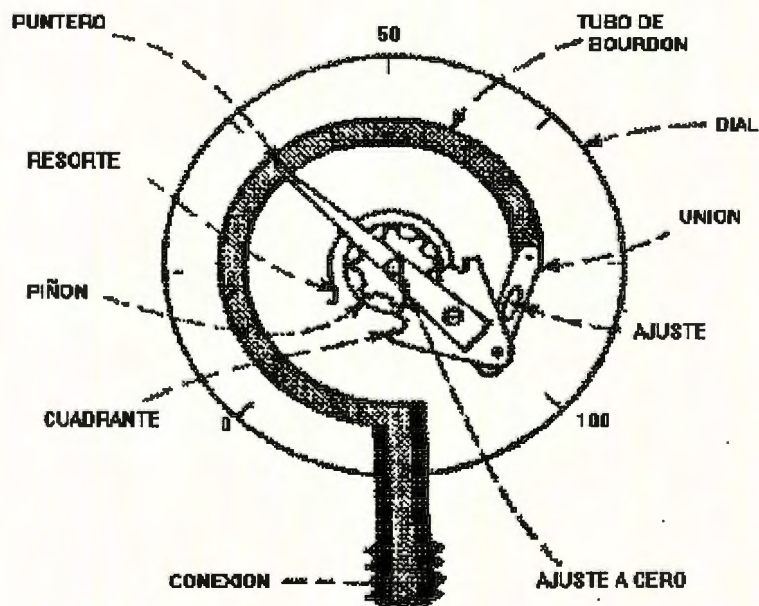


Figura 3. Tubo de Bourdon

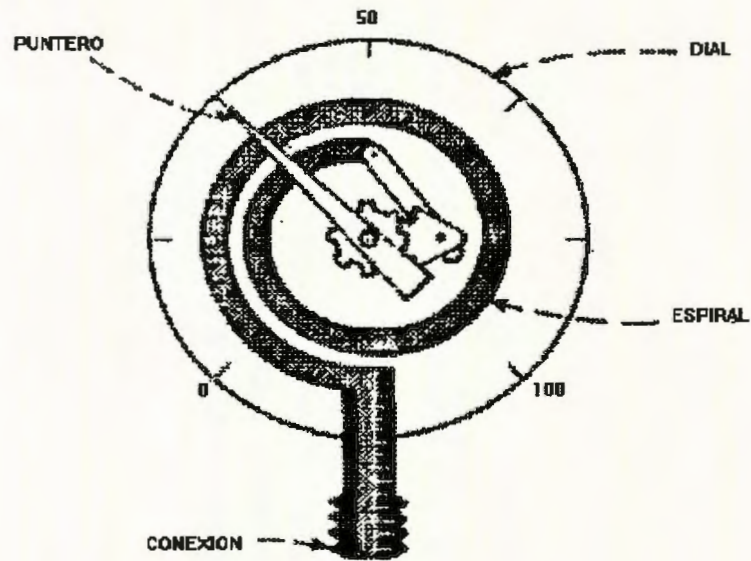


Figura 4. Transductor de presión en espiral

2.1.1.3 Transductor de presión helicoidal

El elemento helicoidal se basa en el mismo principio del tubo de Bourdon y mide presiones manométricas. Cubre rangos muy altos, aproximadamente 80,000 psig en su construcción se utilizan los mismos materiales del elemento que el de espiral. En la figura 5 se presenta un tipo de transductor de presión.

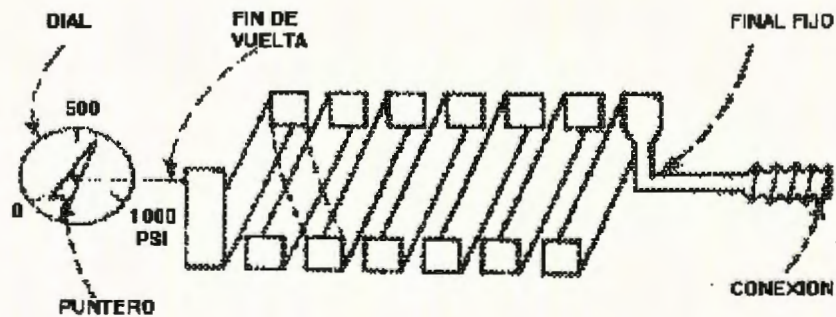


Figura 5. Transductor de presión helicoidal

2.1.1.4 Transductor de presión en fuelle

El fuelle es una pieza flexible axialmente, y puede dilatarse o contraerse con un desplazamiento considerable. Estos elementos se caracterizan por su larga duración. Se utilizan para medir presiones manométricas bajas, no mayores de 30 psig, y también presiones de vacío. El material empleado para el fuelle es usualmente bronce fosforoso, latón o acero inoxidable.

2.1.1.5 Transductor de presión de diafragma

El transductor de presión de diafragma consiste en una o varias cápsulas circulares conectadas rígidamente entre sí por una soldadura, de forma que al aplicar presión cada cápsula se deforma y la suma de pequeños desplazamientos es amplificada por un juego de palancas. El elemento de diafragma se usa para medir presiones manométricas aún más bajas que las que mide el fuelle, también se utiliza para medir presiones de vacío. Los transductores de deformación elástica convierten la presión en un desplazamiento, el cual hay que convertir en una señal eléctrica proporcional. Dos de los métodos más usuales para realizar esta última conversión son los siguientes:

1) Utilización de un LVDT, transductor diferencial de variación lineal, para convertir el desplazamiento en un voltaje alterno. El LVDT es un transductor de desplazamiento cuya salida es un voltaje alterno proporcional a la magnitud del desplazamiento. Este método se utiliza con elementos de tipo Bourdon o fuelle.

2) Colocación de galgas extensiométricas sobre una barra que es sometida a deflexión por un diafragma. El funcionamiento de la galga extensiométrica, se basa en la variación de longitud y de diámetro, y por lo tanto de resistencia, que tiene lugar cuando un hilo de resistencia se encuentra sometido a una tensión mecánica por la acción de una presión.

Existen dos tipos de galga extensiométrica: galgas cementadas, formada por varios bucles de hilo muy fino pegados a una hoja de base de cerámica, papel o plástico, y galgas sin cementar, en la que los hilos de resistencia descansan entre una armazón fija y otra móvil bajo

una ligera tensión inicial. En ambos tipos de galgas, la presión estira o comprime los hilos, modificando la resistencia de los mismos.

La galga forma parte de un puente de Wheatstone, que cuando esta sin tensión tiene una resistencia eléctrica determinada. Se aplica al circuito una tensión nominal tal que la pequeña corriente que circula por la resistencia crea una pequeña caída de tensión en la misma y el puente se equilibra para estas condiciones. Cualquier variación de presión que mueva el diafragma del transductor, cambia la resistencia de la galga y desequilibra el puente.

2.1.2 Transductores de presión piezo-eléctricos

Los elementos piezo-eléctricos son materiales cristalinos que al deformarse físicamente por la acción de una presión generan una señal eléctrica. Son elementos ligeros, de pequeño tamaño y de construcción robusta. Su señal de respuesta a una variación de presión es lineal y son adecuados para medidas dinámicas. Tiene la desventaja de ser sensible a los cambios de temperatura y precisar ajustes de impedancias en caso de fuerte choque. Asimismo su señal de salida es relativamente débil por lo que necesita de amplificadores y acondicionadores de señal que pueden introducir errores en la medición. Dos materiales típicos en los transductores piezo-eléctricos son el cuarzo y el titanato de bario, capaces de soportar temperaturas del orden de 150 °C en servicio continuo y 230 °C en servicio intermitente.

2.1.3 Transductor de presión en circuitos integrados

Los transductores de presión en circuito integrado (IC), contienen dentro de una misma pastilla el sensor, que es un cristal de silicio piezo-resistivo (su resistencia varia con la presión), y el circuito acondicionador de señal que transforma el cambio de resistencia en cambio de voltaje. Este tipo de transductor se usa para medir presiones absolutas, manométricas y diferenciales, y además de su uso en aplicaciones industriales, encuentra muchas otras aplicaciones, como por ejemplo: medir la tensión arterial en los tensiómetros digitales, elementos de medida en los sistemas de control de automóviles basados en

microprocesadores, etc. Los transductores de presión en IC vienen en rangos de presión que abarcan desde cero hasta 5000 psig, y la salida típica es una señal de voltaje continuo entre 2.5 y 12.5 voltios, el cual varía linealmente con la presión de entrada.

El sensor de presión aplicado en este trabajo es el SPX100A (cuyas características se muestran en el anexo A1), el cual es un sensor de presión piezo-resistivo que provee una salida de voltaje proporcional a la presión absoluta aplicada, este elemento se vale de un resistor el cual está integrado a un diafragma de silicio para transformar presiones y fuerzas en una salida eléctrica. El dispositivo está encapsulado en un empaque plástico para mayor protección. En la figura 6 se muestra la construcción física del transductor.

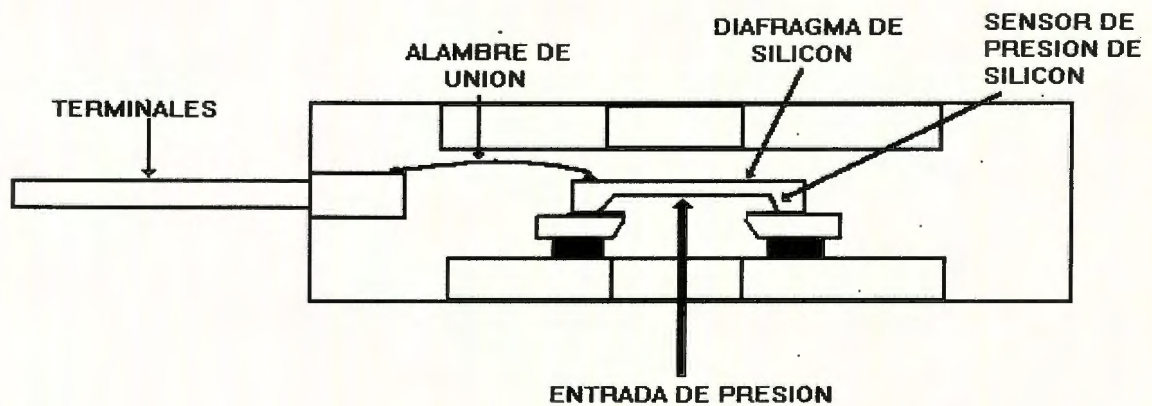


Figura 6. Transductor de Presión de Diafragma Integrado

2.2 TRANSDUCTORES DE NIVEL

Existe un considerable número de transductores de nivel, sin embargo, en esta sección se tratarán los de mayor uso actualmente en la industria.

2.2.1 Transductor de nivel capacitivo

El nivel de un líquido no conductor puede determinarse directamente con una técnica

capacitiva. Este método se basa en la diferencia entre la constante dieléctrica del líquido y la del gas que tiene encima. El sistema se muestra en la figura 7.

La altura del líquido (l) se mide con relación a la altura total (h). Básicamente se dispone de un capacitor dentro del tanque, los electrodos del condensador son dos cilindros coaxiales y su capacidad varía según el nivel del líquido. El transductor convierte el nivel en un valor de capacidad, mediante un puente de corriente alterna, se convierte esa capacidad en un voltaje análogo que se lleva a un amplificador.

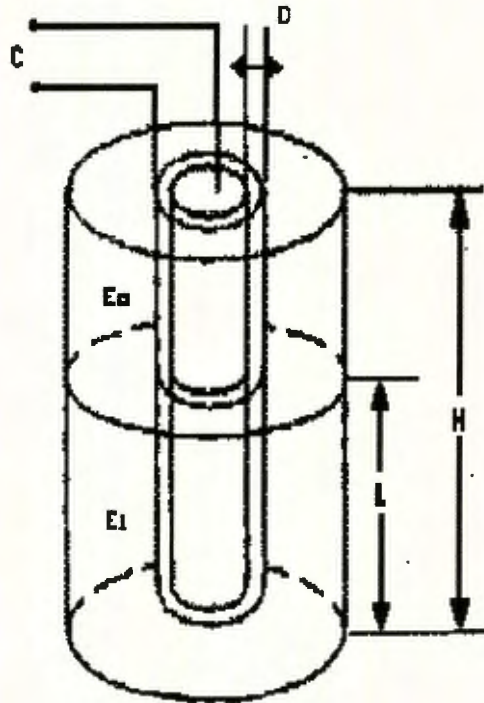


Figura 7. Transductor de nivel capacitivo

2.2.3 Sensor de nivel de electrodos

Este transductor es ocupado con líquidos conductores, es de fácil aplicación y básicamente está constituido por electrodos, de los cuales uno de ellos tiene un voltaje de referencia, y valiéndose de las características conductoras del líquido este voltaje es trans-

mitido a los electrodos, los cuales se encuentran colocados de forma ordenada y ascendente a lo largo del tanque, cuando el líquido aumenta o disminuye su nivel hace contacto físico con los diferentes electrodos, del cual solo se interpreta el del electrodo que se encuentra a nivel más alto por debajo del líquido, de esta forma se conoce el nivel al cual el líquido se encuentra. Este tipo de transductor tiene la desventaja que no puede ser utilizado con líquidos de naturaleza inflamable, ya que existe un alto riesgo de provocar una combustión del líquido, ni con líquidos de alta resistencia eléctrica o aislante, ya que el voltaje no puede ser transmitido a los electrodos, comúnmente este tipo de sensor es utilizado con un buffer, para compensar la caída de voltaje que pueda existir en el líquido, y de esta forma obtener lecturas estables.

Muchas veces surge la necesidad de obtener lecturas de nivel en líquidos de naturaleza inflamables y no conductores, los dos métodos para sensar nivel expuestos anteriormente no satisfacen esta necesidad. Existen métodos muy complicados basados en resonancia los cuales requieren de sensores de alta tecnología y calibración muy precisa. Una alternativa económica y de fácil implementación es el uso de un flotador en conjunto con un transductor de desplazamiento (LVTD) el cual sensa el desplazamiento del flotador que se mantiene en la superficie del líquido.

2.3 SENSORES DE TEMPERATURA

Existen muchos tipos de sensores de temperatura, los dos tipos que se discuten aquí son: los elementos semiconductores, los cuales son utilizados para medir temperaturas en el rango de 55°C a 100°C, y las termocuplas las cuales son utilizadas para medir temperaturas muy bajas y muy altas.

2.3.1 Sensores de temperatura semiconductores

Los dos tipos de sensores de temperatura semiconductores principales son los de

fuentes de voltaje y los de fuente de corriente. Estos sensores incrementan su salida (sea voltaje o corriente, según el tipo de sensor) en forma proporcional a la temperatura sensada por este.

2.3.2 Termocuplas

Cuando dos materiales cualesquiera son puestos en contacto entre sí, un pequeño voltaje es producido entre ellos. El voltaje generado depende de los tipos de metales usados en la unión y además de la temperatura en la superficie de estos. Dependiendo de los metales, el voltaje incrementa entre $7\mu\text{V}$ y $75\mu\text{V}$ por cada incremento de un grado centígrado de temperatura. Diferentes combinaciones de metales son usados para medir diferentes rangos de temperaturas. Una termocupla hecha de unión de hierro y constantano, llamada comúnmente termocupla tipo J es la ocupada en esta aplicación (la hoja de especificaciones se muestra en el anexo A2), esta es de tamaño pequeño y puede tener uso industrial o en laboratorio. Las termocuplas presentan tres problemas principales:

a) El primer problema es el hecho que la salida es muy pequeña y deberá ser amplificada para lograr alcanzar un rango aceptable para lectura.

b) El segundo problema que se presenta es que para obtener medidas precisas, una segunda unión hecha de los mismos metales debe ser incluida en la termocupla para compensar su inestabilidad.

c) El tercer problema con las termocuplas es que su salida de voltaje no cambia linealmente con los cambios de temperatura, esto puede ser corregido con circuitería análoga, la cual cambia la ganancia de un amplificador de acuerdo al valor de la señal.

3. CONVERTIDORES ANALOGO A DIGITAL Y DIGITAL A ANALOGO.

Los sistemas digitales realizan todas sus operaciones internas en binario natural o bien algún tipo de código binario. Cualquier información que vaya a introducirse en un sistema digital debe ponerse en forma binaria antes de que sea procesado por los circuitos digitales.

Muchos dispositivos se usan en el lado de entrada y/o salida de los sistemas digitales para servir como vínculo de comunicación con el mundo externo. Los dispositivos de entrada/salida relacionados con el proceso ofrecen el medio por el cual un sistema digital monitorea y controla un proceso físico. En el lado de la entrada, las dimensiones de los parámetros del proceso que son análogos por naturaleza son generalmente transducidos (cambiados a un voltaje o corriente electrónica proporcional) y enviados a un convertidor de análogo a digital (ADC por sus siglas en inglés), el cual convierte la cantidad analógica en una representación digital correspondiente.

Por lo tanto un convertidor de análogo a digital funciona como interfase entre un dispositivo completamente digital, como una computadora, y el mundo analógico externo.

3.1 METODOS DE CONVERSION DE A/D

Varios tipos de convertidores A/D se encuentran disponibles. Algunos se caracterizan por la velocidad de la conversión, mientras que otros se caracterizan por su alta resolución. Los cuatro métodos utilizados ampliamente son los siguientes:

1. Rampa Simple
2. Rampa Doble
3. Aproximación Sucesiva
4. Flash y Semi-Flash

Cada uno de estos métodos tiene sus ventajas y desventajas. Algunas de las características de estas técnicas de conversión se muestran en la siguiente tabla 1:

TIPO	VELOCIDAD	ERROR	RESOLUCION
RAMPA SIMPLE	LENTA (ms.)	ALTO (± 1 LSB)	MEDIA (8-14 BITS)
RAMPA DOBLE	LENTA (ms.)	ALTO (± 1 LSB)	ALTA (10-18 BITS)
APROXIMACION SUCESIVA	MEDIA (μ s.)	BAJO (± 0.5 LSB)	MEDIA-ALTA (8-16 BITS)
FLASH	ALTA (μ s.)	BAJO (± 0.25 LSB)	BAJA(4--8 BITS)

Tabla 1. Comparación de las características del ADC.

3.2 CRITERIOS DE SELECCION PARA CONVERTIDORES A/D

Para seleccionar un convertidor se deben considerar los siguientes parámetros:

primero, ¿cuantos valores de salida proporciona el convertidor? segundo, ¿cuanto cambiará el bit menos significativo del código de salida al haber un cambio en el voltaje de entrada? tercero, ¿cuantas conversiones se realizarán por segundo?

3.2.1 Número de bits

En un sistema digital (binario) cada bit de información esta representado por un "1" o por un "0". Para representar números del 0 al 15, se requieren cuatro bits. Un número de 8

bits puede representar los números decimales del 0 al 255.

3.2.2 Velocidad de conversión

El proceso de muestreo y conversión de una señal análoga a un número digital no es instantáneo. Se requiere una cantidad de tiempo finito para que el proceso de conversión se complete. Conversiones individuales rápidas resultan en más conversiones por unidad de tiempo. Por ejemplo, una velocidad de conversión de 1.0 ms implica 1000 conversiones por segundo.

3.2.3 Exactitud de la conversión

El convertidor A/D aproxima la curva lineal por una serie de valores de escaleras llamadas niveles de cuantización. Dependiendo de cuando la muestra es tomada, la muestra puede tener error. El método para determinar el error que se puede esperar de un convertidor A/D es expresado de acuerdo al número de niveles de cuantización, específicamente a la variación en el LSB (bit menos significativo).

$$\text{Valor de LSB} = \text{voltaje de escala completa} / 2^n$$

De esta formula, podemos deducir que entre mayor es el número de bits del convertidor A/D menor es el error esperado.

3.3 TECNICAS DE CONVERSION

Las características generales de los convertidores analógico a digital se expusieron en la sección 3.2. Hay tres tipos estándar clasificados según sus tiempos de conversión. El convertidor de rampa normalmente requiere 300 ms para efectuar una conversión. Es la mejor opción para medir voltajes de corriente continua de variación lenta. El siguiente tipo

es el convertidor por aproximación sucesiva el cual tiene la característica de poseer una velocidad media de conversión, una resolución alta y un porcentaje bajo de error. Es una buena elección para medir señales que no varían drásticamente (temperatura, presión, nivel etc.). Los más rápidos de todos son los convertidores más costosos tipo flash los cuales pueden digitalizar señales de vídeo.

3.3.1 Convertidores de aproximación sucesiva

El convertidor de aproximación sucesiva ha sido usado por mucho tiempo y es aún el convertidor A/D más común para aplicaciones con rangos de muestreo de aproximadamente 1 MHz máximo. El nombre de "aproximación sucesiva" viene del hecho de que el convertidor llega a un valor numérico haciendo mediciones sucesivas más precisas hasta que se llega a la respuesta más cercana.

Básicamente el convertidor A/D de aproximación sucesiva, utiliza un convertidor de digital a analógico que permite que los respectivos valores de los bits sean comparados con la entrada, empezando desde el bit más significativo (MSB) hasta el bit menos significativo (LSB), en una serie de pasos. Este método compara en secuencia, una serie de valores binarios con una entrada analógica para producir un código digital de salida en "n" pasos, en donde "n" es la resolución del convertidor. En el pasado se utilizaban escaleras o arreglos de resistencias para implementar la conversión. Técnicas recientes de diseño emplean redes de capacitores conmutados utilizando redistribución de carga en los capacitores para completar la conversión A/D. En la figura 8 se muestra el diagrama a bloques de un convertidor A/D de aproximación sucesiva típico.

En la salida, un comparador de voltaje simple es manejado por la señal en una entrada y por una referencia del convertidor D/A en la otra. El convertidor de D/A es programado (con MSB = 1, otros bits = 0) para producir un voltaje igual a la mitad del rango medido de la lectura de escala completa del convertidor A/D. El comparador determina si la salida D/A esta arriba o abajo del nivel de la señal de entrada. El DAC (convertidor de digital a analo-

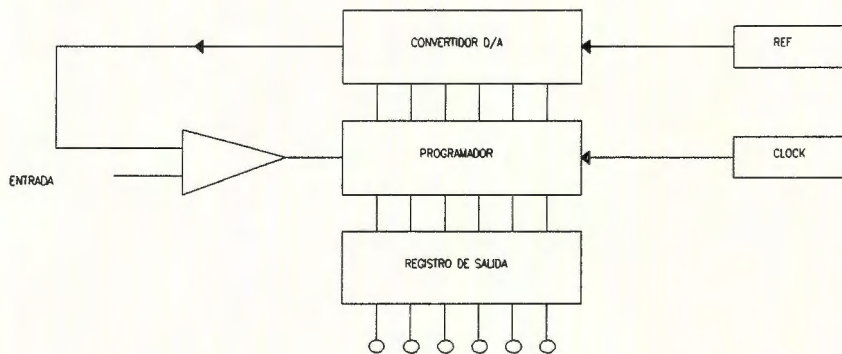


Figura 8. Diagrama a bloques de un convertidor de aproximación sucesiva

análogo) incrementa la salida por $1/2$ de la cantidad previa y dependiendo de como está la entrada, el comparador genera el segundo dígito del código. Este proceso continua hasta que el paso anterior ya no pueda ser dividido por mitad. El resultado es guardado en el registro de salida.

Mientras todo esto sucede rápidamente, la señal debe ser mantenida constante en tanto que el convertidor de aproximación sucesiva hace su proceso. Esto se logra con un circuito de muestra y retención (sample and hold) el cual puede estar adelante del comparador. Entre conversiones, el circuito de muestra y retención adquiere la señal de entrada, y justo antes de que la conversión se inicie esta es puesta en memoria, donde se mantiene hasta que la conversión se completa.

Básicamente un circuito sample and hold es un dispositivo memorizador de voltaje que almacena un voltaje dado en un capacitor de alta calidad. Este capacitor debe tener baja fuga y baja absorción de dieléctrico. Un switch electrónico es conectado al capacitor de tal forma que cuando el switch se cierra, el capacitor se carga con el voltaje de entrada. Cuando el switch se abre, el capacitor retiene esta carga por el período de tiempo deseado.

3.3.1.1 Asignación de pesos por bit de un número binario

En un número binario, el bit más significativo es siempre igual a la mitad de el voltaje de escala completa o valor máximo. El bit más significativo siguiente es igual a 1/4 del valor máximo el siguiente 1/8, el siguiente 1/16 y así sucesivamente hasta que se llega al bit menos significativo el cual tiene el peso de valor máximo/ 2^n (en donde n = al número de bits).

Como un ejemplo, asuma que el voltaje de escala completa (o valor máximo) es de 4V. Con un convertidor de 4 bits, un número binario de 4 bits podría ser representado como sigue:

MSB (Bit más significativo)	= $4 \times 1/2 = 2 \text{ V}$
Siguiente MSB	= $4 \times 1/4 = 1 \text{ V}$
Siguiente MSB	= $4 \times 1/8 = 1/2 \text{ V}$
LSB (Bit Menos Significativo)	= $4 \times 1/16 = 1/4 \text{ V}$
Lectura de escala completa	= $3 \text{ } 3/4 \text{ V}$

Por lo tanto en un convertidor de 4 bits la resolución esta definida como:

resolución = escala completa / 2^4 bits = 1/4 V por lo tanto se puede determinar el valor actual en pasos de 1/4 de voltios.

3.3.1.2 Creando un número digital

Como se estableció previamente un convertidor de aproximación sucesiva tiene un convertidor de D/A el cual compara su salida con la entrada análoga. A continuación se escoge el número binario que se aproxima a la entrada analógica. Para hacer esto el convertidor A/D empieza con el bit más significativo ver figura 9. Como se describió anteriormente el A/D compara la mitad del valor de escala completa con la entrada análoga y decide si este esta en el valor, arriba o abajo. Si la entrada está abajo, entonces se fija un "0" en el MSB, si la entrada esta en el valor o arriba del valor de la mitad del voltaje de escala completa, se fija un "1". A continuación se considera el siguiente bit más

significativo de la misma forma, y el proceso continúa hasta que se ha chequeado el bit menos significativo (LSB). Después de chequear el LSB el número que permanece en el registro es la representación binaria más exacta del voltaje análogo de entrada.

Para el convertidor de 4 bits, únicamente se requieren 4 pasos para llegar al número digital. Para un convertidor de 8 bits, se requerirían 8 pasos. El número de bits determina el número de pasos requeridos.

3.4 TIPOS DE DISPOSITIVOS

El uso de técnicas digitales en las comunicaciones, mediciones y sistemas de control se ha incrementado aceleradamente en pocos años. Los sistemas de control utilizan una combinación de técnicas de circuitos digitales y analógicos. Uno de los dispositivos claves en estos sistemas es el convertidor de análogo a digital. Debido a su complejidad, los convertidores A/D han sido siempre un reto para los diseñadores de circuitos integrados.

A través del progreso continuo dispositivos monolíticos económicos se encuentran disponibles para satisfacer un amplio rango de aplicaciones. Hoy en día, la tecnología LinCMOS® provee incrementos de velocidad, mejor estabilidad en la salida, menores requerimientos de potencia, y una estabilidad térmica para las aplicaciones que lo demanden.

3.4.1 ADC0809 convertidor CMOS análogo a digital con multiplexor de 8 canales

El ADC0809 es un convertidor A/D de 8 bits que posee entradas de direcciones así como salidas de tres estados mantenidas (latched). El error del ADC0809 es de $1.0 \pm \text{LSB}$. Este dispositivo tiene un tiempo de conversión de 0.1 ms, puede ser alimentado con una fuente de 5V y puede ser conectado fácilmente a un microprocesador.

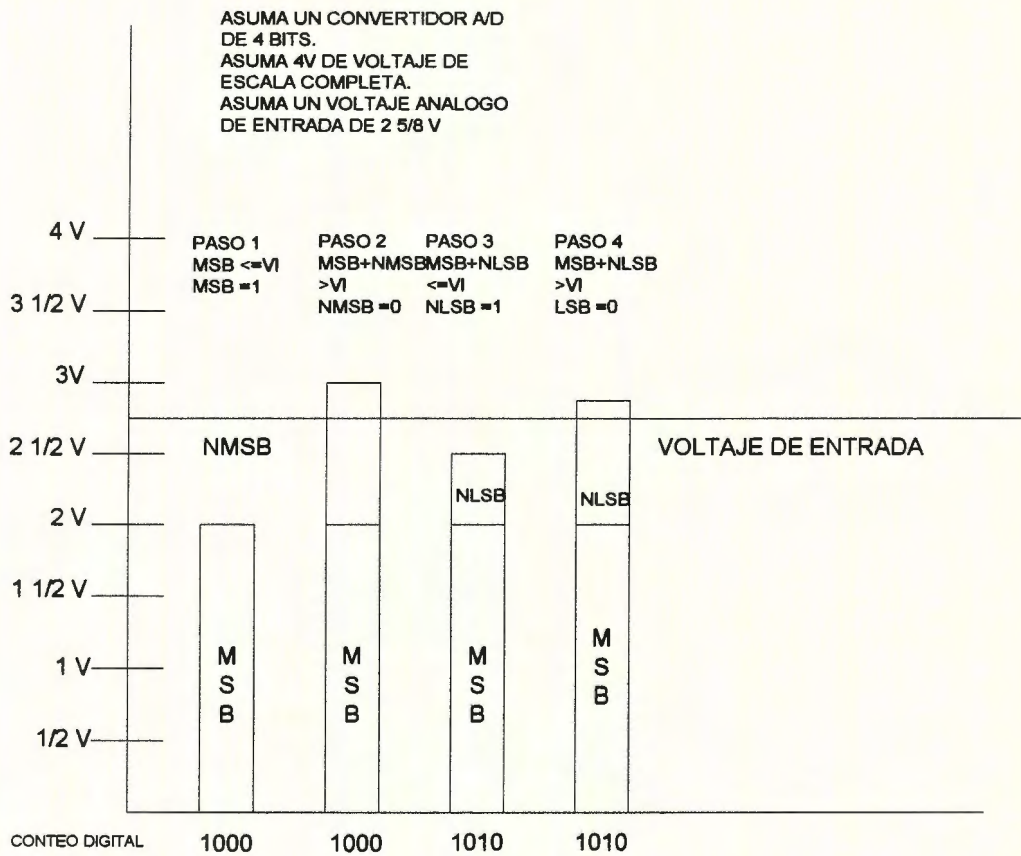


Figura 9. Formación de un número digital

3.4.1.2 Descripción

El ADC0809 es un dispositivo monolítico CMOS con un multiplexor de 8 canales, un convertidor de A/D de 8 bits, y una lógica de control compatible con un microprocesador. El multiplexor de 8 canales puede ser controlado por un microprocesador a través del decodificador de direcciones de 3 bits con un control de carga de direcciones (address load control) para seleccionar cualquiera de las 8 entradas conmutadas que se conectan directamente al convertidor. El convertidor de 8 bits utiliza la técnica de conversión de aproximación sucesiva que se caracteriza por tener un detector de umbral de alta impedancia, un arreglo de capacitores, y un registro de aproximación sucesiva de muestra y

retención (sample and hold register). El diagrama de bloques funcional se muestra en el anexo A3 y la secuencia de operación se muestra en el anexo A4.

Los métodos de conversión utilizados, eliminan la posibilidad de códigos perdidos, no monotonía, y la necesidad de ajuste de cero o escala completa. Los bajos requerimientos de potencia y la alimentación de una fuente simple de 5V hacen que el ADC0809 sea útil para una amplia variedad de aplicaciones. Los componentes principales del ADC0809 son:

Multiplexor: el multiplexor selecciona una de 8 entradas o canales simples determinadas por el decodificador de direcciones. Address Load Control, carga el código de la dirección en un decodificador en una transición de bajo a alto.

Convertidor: el detector de umbral CMOS en el sistema de conversión de aproximación sucesiva determina cada bit examinando la carga en una serie de capacitores con pesos binarios.

En la primera fase del proceso de conversión, la entrada análoga es muestreada cerrando unos switchs electrónicos, y cargando simultáneamente todos los capacitores con el voltaje de entrada. En la siguiente fase del proceso de conversión, todos los switch son abiertos y el detector de umbral empieza a identificar bits, comparando el voltaje de carga en cada capacitor con el voltaje de referencia. En la secuencia de conmutación, los 8 capacitores son examinados separadamente hasta que los 8 bits son identificados y luego la secuencia de conversión de carga es repetida. Con cada paso del proceso de muestreo de capacitores, la carga inicial es redistribuida entre los capacitores. Este proceso de conversión es de aproximación sucesiva.

3.5 CONVERTIDOR DIGITAL ANALOGO.

Los conceptos discutidos, en la sección de análogo a digital se aplican para la conversión de digital a análogo, por lo tanto no serán tocados en esta sección. Se hará

referencia únicamente al método de conversión y al dispositivo utilizado. Un convertidor de digital a análogo, acepta un valor numérico (usualmente un número binario), y genera un voltaje análogo cuyo valor es directamente proporcional a ese valor numérico. El convertidor utilizado consiste en un MSI de 8 bits con entrada compatible CMOS, TTL de alto voltaje y es totalmente compatible con un microprocesador a través de un control lógico. El convertidor contiene una red escalonada de resistencia con pesos binarios, un amplificador de salida de alta velocidad, una referencia de precisión de alto voltaje, y una lógica de control que permite al D/A ser conectado directamente a buses de datos de 8 o 16 bits.

El sistema que controla la corriente en este controlador es llamado R-2R escalonada, una red escalonada, se muestra en la siguiente figura y se llama así porque utiliza solo dos valores de resistencia, un valor llamado R y dos veces el valor de R o 2R. El escalonador es la parte más importante del controlador, su funcionamiento está basado en la división igual de corriente, en cada enlace de resistencia (puntos A,B,C). Estos están ilustrados en la figura 10 .

Este proceso de dividir la corriente de rama previa por la mitad a cada nuevo punto de enlace continua hasta cada punto de enlace. Dado que la corriente es dividida por dos la corriente del primer punto de enlace es la mitad de la corriente de entrada, en el segundo punto es un cuarto de la corriente de entrada, en el tercer punto es un octavo y así sucesivamente.

La salida del convertidor D/A depende de dos condiciones obvias: la magnitud de la entrada codificada en binario de 8 bits y la ganancia del amplificador operacional. Existe también una tercera consideración que controla la salida máxima del convertidor D/A. Es el voltaje de alimentación dado que el IC está construido para funcionar con una amplia gama de voltaje de alimentación, de 4.5 a 16.5, hay dos gamas de salida seleccionadas por el usuario: 0 a 2.56 voltios y 0 a 10 voltios. La gama de salida debe ser siempre inferior al voltaje de alimentación empleado. No puede emplear una gama de salida de 10 voltios si el

voltaje de alimentación solo fuera de 5 voltios. Si emplea la gama de 10 voltios el voltaje de alimentación debe ser de al menos 11.4 voltios, y por lo menos 4.5 voltios para la gama de 2.56 voltios.

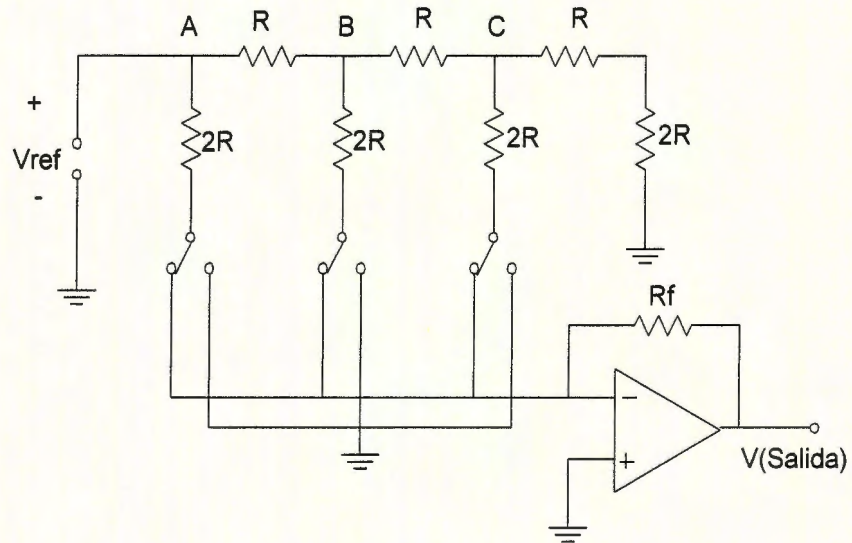


Figura 10 Convertidor R-2R

4. DISPOSITIVOS DE INTERFASE

El incremento en el uso de las computadoras, microprocesadores y elementos lógicos de control, ha generado un reto para la transmisión digital de datos de un sistema a otro. Las comunicaciones digitales entre componentes de sistemas en máquinas de control en una planta, necesita de manejadores de línea que se encarguen de amplificar y regenerar las señales a ser procesadas. Previo a la introducción de los datos al computador se necesita de dispositivos controladores y almacenadores para el manejo de estas señales.

4.1 MANEJADORES DE LINEA

Probablemente la aplicación mas común en un sistema digital, es la transmisión de datos de una compuerta lógica directamente a otra. Las compuertas TTL estándar pueden operar a frecuencias hasta de 20 MHz. Sin embargo, las interconexiones deben ser cortas (pocas pulgadas), y se debe tener cuidado especial de asegurar adecuados márgenes de ruido y reflexiones mínimas de línea. El manejo directo de una compuerta tipo TTL tiene varios factores limitantes. Las corrientes de salida son muy pequeñas para líneas largas. Otro factor a considerar es el uso de compuertas en un ambiente con un alto nivel de ruido. Es necesario el uso de un manejador de línea el cual tiene la función de regenerar la señal y amplificarla para que pueda viajar distancias mayores. Otra de las funciones de los manejadores de línea es la de conmutar, debido a esto es de gran utilidad para el manejo de los diferentes buses de un sistema de computadora.

4.2 EL SN74LS244 / SN74LS241

El SN74LS244 es un buffer con salida de tres estados que puede ser utilizado como

manejador de línea (line driver). El IC contiene ocho buffers diseñados para satisfacer las necesidades de manejar transmisores y receptores, manejadores de direcciones de memoria de tres estados, manejadores de clocks, etc. Estos dispositivos se caracterizan por poseer un alto fan out, un fan in mejorado, y 400 mv de margen de ruido. Además la familia SN74' está diseñada para operación en rangos de temperatura de 0 ° C a 70 ° C (ver anexo A5).

4.3 DECODIFICADORES/DEMÚLTIPLEXORES

Un decodificador/demultiplexor es un circuito lógico que convierte un código binario de entrada de “n” bits en “m” líneas de salida, tal que cada una de estas líneas de salida sea activada solo para una posible combinación de entradas , en tales dispositivos existen 2^n posibles códigos de entrada , para cada una de estas combinaciones solo una de las “m” salidas será alta o baja activa.

Algunos decodificadores/demultiplexores tienen una o más entradas habilitables que se utilizan para controlar la operación del decodificador.

4.3.1 El decodificador/demultiplexor 74LS138

El código de entrada se aplica en A, B y C (ver anexo A6) donde C es el más significativo. Con tres entradas y ocho salidas, este es un decodificador comúnmente llama-

G1	G2	G3	SALIDAS
0	0	1	RESPONDER AL CÓDIGO DE ENTRADA SELECCIONADO EN A, B y C
1	X	X	SALIDA DESACTIVADA
X	1	X	SALIDA DESACTIVADA
X	X	0	DECODIFICADOR DE 1 A8

Tabla 3. Tabla de códigos de salida del 74LS138

do de 3 a 8. Las entradas G1, G2 y G3 son entradas separadas con el fin de activar las salidas que han sido seleccionadas mediante el código de entrada, la operación de este dispositivo se resume en la siguiente tabla de verdad (tabla 2).

Como un demultiplexor toma un código de entrada y lo distribuye en varias salidas, la fuente de datos de entrada se distribuye selectivamente. Para conectar un dispositivo de entrada/salida con un microprocesador, es necesario decodificar la dirección del microprocesador para hacer que el dispositivo funcione como una sección única del mapa de memoria. Este procedimiento es explicado en la siguiente tabla (tabla 3). Sin un decodificador de direcciones, únicamente un dispositivo de entrada/salida podría ser conectado al microprocesador lo cual lo dejaría subutilizado.

Entre más líneas de dirección se utilicen en un decodificador, se generan menos direcciones imágenes.

DIRECCION EN DECIMAL	FUNCION	SA9	SA8	SA7	SA6	SA5	SA4	SA1	SA0
608	PUERTO A	1	0	0	1	1	0	0	0
609	PUERTO B	1	0	0	1	1	0	0	1
610	PUERTO C	1	0	0	1	1	0	1	0
611	WORD CONTROL	1	0	0	1	1	0	1	1
	74LS138	6	5	3	2	1	4	A1	A0

Tabla 3. Tabla de conexión del 74LS138 y el slot.

Las líneas de SA0 a SA9 proceden del bus de direcciones del microprocesador de la computadora donde SA0 es la menos significativa y SA9 es la mas significativa. A0 y A1 activan las líneas que seleccionan los registros dentro del PPI. Las líneas 6,5 y 4 activan el input enable del 74LS138, las líneas 3,2 y 1 activan el input select del 74LS138, la salida en el pin 12 del 74LS138 activa el CS del PPI, para que solamente responda a las direcciones 608,609,610, y 611(en decimal).

El dato a escribir como palabra de control en el PPI es 152 el cual fija a este dispositivo en modo cero, configurando el PPI en el siguiente modo: puerto A como entrada, puerto B como salida, la parte baja del puerto C como salida y la parte alta de C como entrada. De tal manera que en el puerto A se introduzcan los datos a monitorear; en el puerto B se ejerza el control en el sistema mediante la salida de datos, y en el puerto C se maneje el protocolo de comunicación que controla el ADC0809 y el convertidor de digital a análogo.

4.4 EL PPI 8255 A

El 8255A es un interfase de periféricos programable (PPI) de entrada/salida que tiene como función interconectar un equipo al sistema de bus de una computadora, su configuración es programable por medio de software. El PPI8255A esta compuesto por los elementos que a continuación se describen.

4.4.1 Bus bidireccional de datos

Este dispositivo contiene tres puertos los cuales pueden ser programados para operar en uno de los tres modos de operación que posee. Los diferentes modos que posee permiten al programador que utilice el dispositivo para la transferencia de datos en formato paralelo de 8 bits bidireccional de tres estados. Las palabras de control y el estatus de la información son también transferidos a través del manejador del bus de datos.

4.4.2 Lógica de control de lectura/escritura

La función de este bloque es manejar todas las transferencias internas o externas de datos de control o palabras de estado. Este acepta la señal de entrada del direccionamiento del CPU y de los buses de control y a cambio emite comandos para ambos grupos de control, este bloque esta compuesto por los siguientes pines:

Selección de chip (cs): un bajo en este pin de entrada permite la comunicación entre el 8255 y el CPU.

Lectura (rd): un bajo en este pin de entrada permite que el 8255 envíe el dato o el estado de información al CPU a través del bus de datos. Esto significa que permite que el CPU lea de la PPI8255A.

Escritura (wr): un bajo en este pin de entrada permite que el CPU escriba datos o palabras de control en la PPI8255A.

Direcciones internas de entrada (A0 y A1): estas señales de entrada en unión con las señales de lectura y escritura controlan la selección de uno de los tres puertos y del registro de control de acuerdo a la siguiente tabla de la verdad.

Reset : un alto en esta línea de entrada permite que todos los buses de los puertos sean inicializadas como entradas. Esto previene la destrucción de circuitos conectados a los puertos. También limpia el registro de control.

A1	A0	CONTROL
0	0	PORT A
0	1	PORT B
1	0	PORT C
1	1	CONTROL WORD

Tabla 4. Tabla de selección de registro del PPI.

4.4.3 Operación básica

La operación básica de este circuito se lleva a cabo mediante el control de sus registros respondiendo a la siguiente tabla (tabla 5).

4.4.4 Características de cada puerto

PUERTO A: Salida de 8 bits con Buffer/Latch y entrada de 8 bits con Latch.

PUERTO B: Entrada de 8 bits con Latch y Salida Buffer de 8 bits.

A1	A0	RD	WR	CS	OPERACION DE ENTRADA (LECTURA)
0	0	0	1	0	PUERTO A Bus de Datos
0	1	0	1	0	PUERTO B Bus de Datos
1	0	0	1	0	PUERTO C Bus de Datos
A1	A0	RD	WR	CS	OPERACION DE SALIDA (ESCRITURA)
0	0	1	0	0	Bus de Datos PUERTO A
0	1	1	0	0	Bus de Datos PUERTO B
1	0	1	0	0	Bus de Datos PUERTO C
1	1	1	0	0	Bus de Datos PALABRA DE CONTROL
A1	A0	RD	WR	CS	FUNCION DE DESHABILITACION
X	X	X	X	1	Bus de Datos 3 ESTADOS
1	1	0	1	0	CONDICION ILEGAL
1	X	1	1	0	Bus de Datos 3 ESTADOS

Tabla 5. Operación básica del PPI.

PUERTO C: 8 bits de salida Latch/Buffer y 8 bits de entrada con Buffer. Este puerto puede ser dividido en 2 puertos de 4 bits según el modo de operación seleccionado. Cada puerto contiene cuatro Latch y pueden ser usados para controlar la señal de salida y la señal de estado en conjunto con los puertos A y B.

Para mayor comprensión se muestra el diagrama interno a bloques del PPI8255A en la figura 11.

4.4.5 Modos de operación e inicialización.

La PPI puede ser programada por medio de software para una situación en particular en la que se requiera que ciertos pines lleven a cabo la función de entrada y otros lleven a cabo la función de salida. Existen modos de operación determinados en los cuales cada

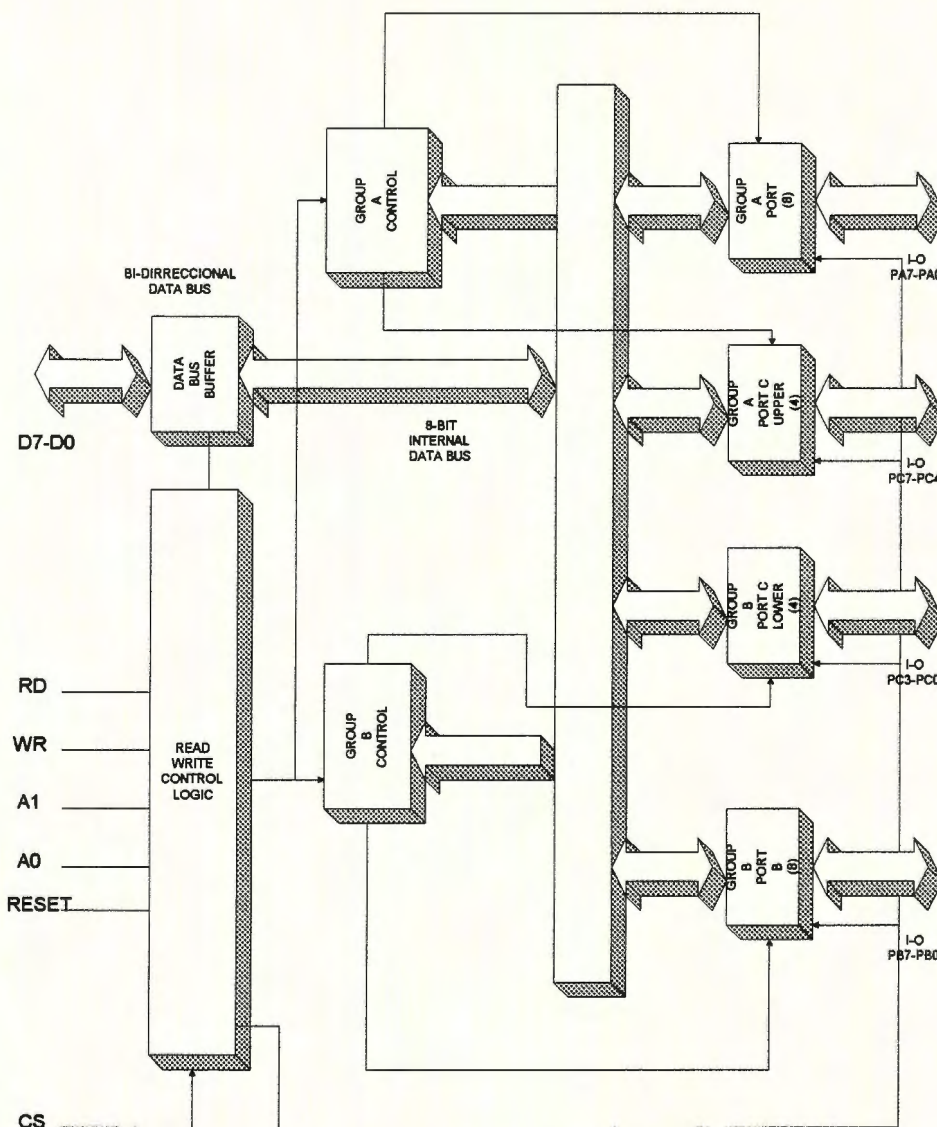


Figura 11. Diagrama a bloques del PPI

puerto puede funcionar de forma individual aislada o en conjunto para poder tener opción de comunicación por handshaking. Los tres modos de operación determinados (Modo 0, Modo 1, Modo 2) se describen a continuación.

4.4.5.1 Modo 0.

Este es utilizado cuando se quiere usar un puerto como simple entrada o salida de

datos, sin Handshaking . Si ambos puertos A y B son inicializados en modo cero, entonces las dos mitades del puerto C pueden ser usadas juntas como un puerto adicional de 8 bits, o usados individualmente como dos puertos de cuatro bits, en tal caso una parte puede ser utilizada como entrada y la otra como salida independientemente entre ellos. En resumen este modo provee de entradas o salidas simples a cada uno de los tres puertos. Este modo no requiere ajustes y los datos son escritos o leídos del puerto específico. En este modo existen dos puertos de ocho y dos puertos de cuatro bits; cualquier puerto puede ser entrada/salida; las salidas son mantenidas (Latched); las entradas no son mantenidas; para este modo existen 16 configuraciones posibles, que se detallan mediante una tabla (ver anexo A7). Este modo es utilizado en el sistema debido a que el protocolo de comunicación se maneja a través de software.

4.4.5.2 Modo 1

Cuando se quiere usar el puerto A o el puerto B para un Handshaking en una operación de entrada o salida, se inicializa el puerto en modo 1. En este modo algunos de los pines del puerto C funcionan como líneas de handshaking, esto es los pines PC0, PC1, y PC2 funcionan como líneas de handshaking para el puerto B cuando es salida y los pines PC3, PC6 y PC7 si el puerto A es también salida; los pines restantes PC4 y PC5 funcionan como entrada o salida. Cuando el puerto A o B son salidas las señales que manejan el protocolo se identifican como: buffer de salida lleno (OBF), entrada reconocida (ACK) y petición de interrupción (INTR). Para el puerto A en modo uno como entrada los pines PC3, PC4 y PC5 funcionan como señales de handshaking y para el puerto B como entrada corresponden los pines PC0, PC1 y PC2 ; y los PC6 y PC7 están disponibles para ser usados como líneas de entrada o salida. Las señales que manejan el protocolo cuando A y B son entradas son: strobe (STB), buffer de entrada lleno (IBF) y petición de interrupción (INTR). En resumen este modo provee un método de transferencia de entrada/salida desde/hacia un lugar con señales de protocolo. En este modo los puertos A y B usan las

líneas del puerto C para aceptar y generar las señales del protocolo. Aquí cada puerto (A y B) posee bits de control del puerto C, cada dato de ocho bits pueden ser de entrada o salida y además sostenidos (latched), el puerto C es usado como de control y estado.

4.4.5.3 Modo 2

En esta configuración solo el puerto A puede ser inicializado en modo 2, es decir el puerto A puede ser usado para un handshaking bidireccional en la transferencia de datos. Esto significa que los datos pueden ser de entrada como de salida en el mismo bus de ocho bits; entonces en este caso los pines PC3-PC7 son usados como líneas para handshaking del puerto A. Los otros tres pines del puerto C pueden ser usados para entrada o salida si el puerto B esta en modo cero o pueden ser usados por el puerto B como líneas de handshaking si el puerto B es inicializado en modo 1. Este modo provee un medio para la comunicación con un dispositivo periférico en una estructura de ocho bits para la transmisión y recepción de datos. El protocolo maneja la transferencia de información en forma ordenada mediante las siguientes señales: petición de interrupción (INTR), buffer de salida lleno (OBF), entrada o salida reconocida (ACK), strobe (STB) y buffer de entrada lleno (IBF). En este modo la generación de interrupciones y habilitación como deshabilitación están disponibles, además solo se utiliza el puerto A, en forma bidireccional y cinco bits del puerto C mantienen el control y manejan el protocolo, las entradas de datos como de salidas del puerto son sostenidas (latched) .

La figura 12 muestra como se pueden construir las palabras de control para los distintos modos de operación del PPI8255A.

4.5 DIRECCIONAMIENTO DE LA COMPUTADORA HACIA EL PPI8255A A TRAVES DEL SLOT.

La computadora es un sistema que se puede describir como una unidad de

procesamiento, manejo y almacenamiento de datos, la cual realiza todas sus tareas valiéndose de un componente llamado microprocesador para realizar todas las funciones de direccionamiento, control y manejo de la información hacia todos los dispositivos que interactúan con ella.

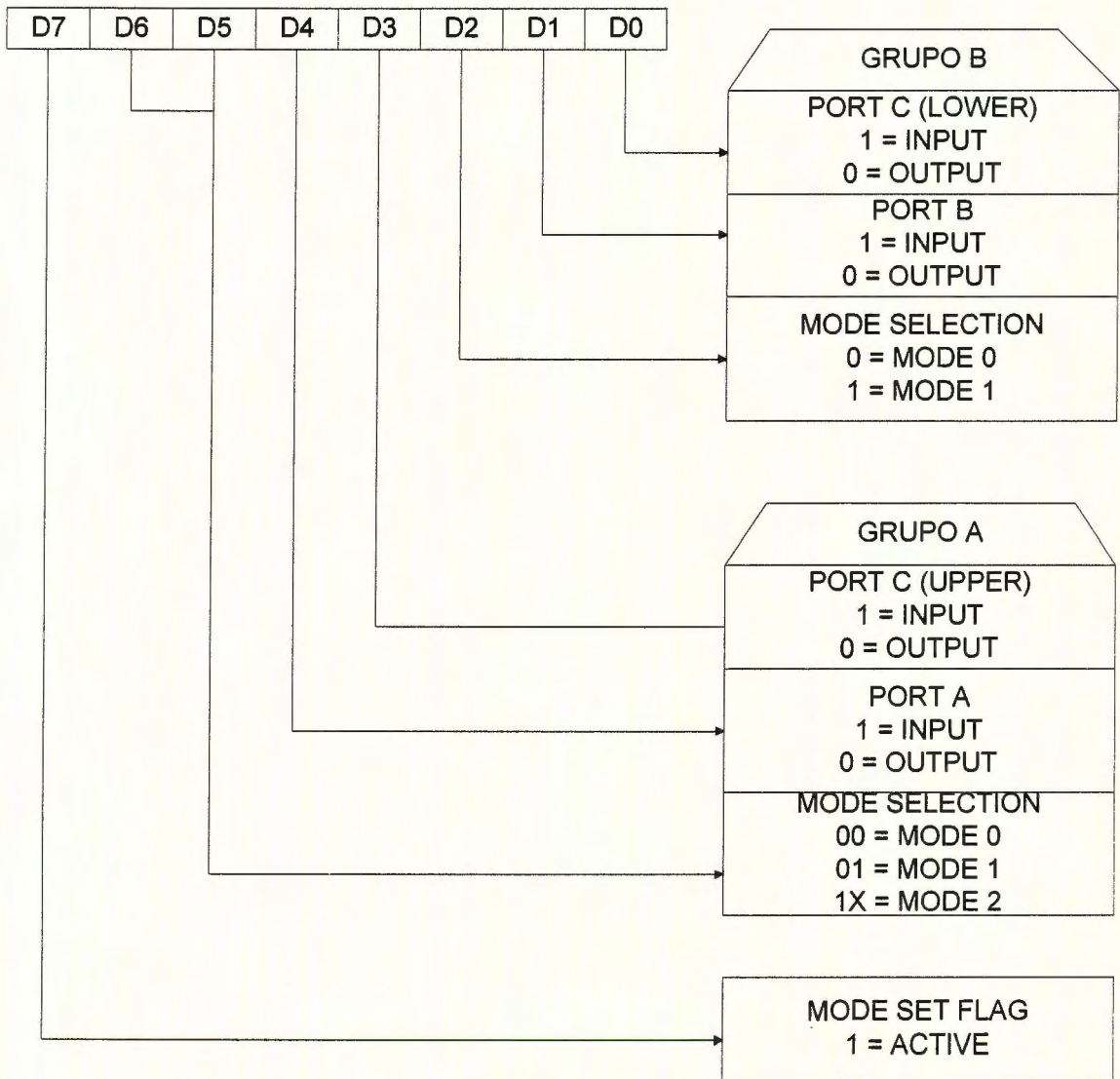


Figura 12. Ensamble de la palabra de control

Por lo tanto para poder manejar la interfase que controla la adquisición de datos es

necesario obtener las direcciones en donde se encuentran los datos que se desean acceder. Este direccionamiento se hace conectando la interfase de adquisición de datos a la computadora por medio de uno de los slots que contiene la computadora para conectarse al mundo exterior. Este procedimiento se hace interconectando las líneas que interactúan de la siguiente manera:

1. Las líneas READ, WRITE, RESET, Vcc y GND del microprocesador a sus correspondientes en el PPI8255A para entrelazar la comunicación entre el microprocesador y el PPI.

2. Las ocho líneas menos significativas del bus de datos del Microprocesador a las líneas correspondientes del bus de datos del PPI para hacer la transferencia de información.

3. El address enable y las líneas del bus de direcciones del microprocesador al decodificador 74LS138, donde las dos líneas menos significativas del address bus del microprocesador se conectan a las líneas de address port del PPI para realizar así la selección de los registros en el PPI mediante los pines A0 y A1; obteniéndose así una decodificación adecuada que pueda ser interpretada correctamente a través del software.

5.0 ETAPA DE CONTROL

Una de las áreas en las cuales el microprocesador y las computadoras han tenido mayor impacto es el control de procesos industriales. El control de procesos envuelve primero la medición de variables del sistema tales como: la temperatura, la velocidad de un motor, el nivel de un líquido en un tanque, el grosor de un material, etc. El sistema es entonces ajustado hasta que el valor de cada variable es igual a un valor predeterminado para esa variable, a este valor se le llama usualmente punto de fijación (set point). El sistema de control debe mantener cada variable tan cerca como sea posible al valor de set point, y debe compensar tan rápida y exactamente como sea posible cualquier cambio en el sistema, tal como un delta de variación en la variable que se controla. El control de procesos tiene amplia aplicación en la industria . En estos sistemas se usan con frecuencia controles programados como el de la temperatura de un horno de calentamiento que se controla según un programa preestablecido. Este programa puede consistir en elevar la temperatura a un determinado valor durante un intervalo de tiempo, y luego reducir a otra temperatura prefijada también durante un intervalo de tiempo predeterminado. El controlador, entonces funciona manteniendo la temperatura del horno al punto de ajuste variable.

5.1 SISTEMA DE CONTROL DE LAZO CERRADO

Con frecuencia se llama así a los sistemas de control retroalimentado. En la practica, se utilizan indistintamente el control retroalimentado o control de lazo cerrado, la señal de control actuante, que es la diferencia entre la señal de entrada y la retroalimentación (que puede ser la señal de salida o una de las funciones de la señal de salida y sus derivados), entra al controlador para reducir el error y llevar la salida del sistema a un valor deseado. El

termino lazo cerrado implica siempre el uso de la acción de control retroalimentado para reducir el error del sistema.

5.2 SISTEMA DE CONTROL DE LAZO ABIERTO

El sistema de control en el que la salida no tiene efecto sobre la acción de control, se denomina sistema de control de lazo abierto. En otras palabras, en un sistema de control de lazo abierto la salida ni se mide ni se realimenta, para compararla con la entrada. Un ejemplo práctico la constituye una lavadora de ropa doméstica, donde la máquina no mide la limpieza de la ropa. En un sistema de control de lazo abierto la salida no se compara con la entrada de referencia. Por lo tanto, para cada entrada de referencia corresponde una operación de condición fija. Así, la precisión del sistema depende de la calibración. En presencia de perturbaciones, un sistema de control de lazo abierto no cumple su función asignada. En la práctica el control de lazo abierto solo se puede utilizar si la relación entre la entrada y la salida es conocida, y si no se presentan perturbaciones tanto internas como externas.

5.3 ACCIONES BÁSICAS DE CONTROL Y CONTROLADORES AUTOMÁTICOS INDUSTRIALES

Un controlador automático compara el valor real de la salida de una planta con la entrada de referencia (valor deseado), determina el error, y produce una señal de control que reducirá el error a cero, o a un valor muy pequeño. La forma como el controlador automático produce la señal de control, se denomina acción de control.

5.3.1 Acciones Básicas de control

Los controladores analógicos industriales se clasifican según su acción de control en:

1. controladores de dos posiciones, o intermitentes (encendido o apagado).
2. controladores proporcionales.
3. controladores integrales.
4. controladores proporcional-integral.
5. controladores tipo proporcional-derivativo.
6. controladores tipo proporcional-integral-derivativo.

La mayoría de los controladores analógicos industriales utilizan electricidad o algún fluido, como aceite o aire a presión, a modo de fuentes de potencia. Los controladores analógicos también se pueden clasificar según el tipo de potencia que utilizan en su operación, como neumáticos, hidráulicas o electrónicos. La clase de controlador a usar se decidirá en base a la naturaleza de la planta y las condiciones de operación, incluyendo consideraciones tales como seguridad, costo, disponibilidad, confiabilidad, exactitud, peso y tamaño. Debido a que nuestros dispositivos de control son de dos posiciones, se centrará el estudio en los de este tipo.

En un sistema de control de dos posiciones, el actuador tiene solo dos posiciones fijas, que en muchos casos son simplemente conectado y desconectado. El controlador de dos posiciones, o de encendido-apagado es relativamente simple y económico, y por esta razón se usa ampliamente en sistemas de control, tanto industriales como domésticos.

Sea $U(t)$ la señal de salida del controlador y $e(t)$ la señal de error. En un controlador de dos posiciones, la señal $U(t)$ permanece en un valor máximo o mínimo, según sea la señal de error positiva o negativa, de manera que:

$$u(t) = U_1 \text{ para } e(t) > 0$$

$$u(t) = U_2 \text{ para } e(t) < 0$$

donde U_1 y U_2 son constantes. Generalmente el valor mínimo de U_2 puede ser, o bien cero o bien $-U_1$. En general los controladores de dos posiciones son dispositivos eléctricos, donde habitualmente hay una válvula accionada por un solenoide eléctrico. Los controladores neumáticos proporcionales con muy alta ganancia también actúan como

controladores de dos posiciones y se les conoce como controladores neumáticos de dos posiciones. En las figuras 13a y 13b se pueden ver diagramas de bloques de controladores de dos posiciones. El rango en la que la señal de error debe variar antes que se produzca la conmutación, se denomina brecha diferencial o zona muerta. En la figura 13b se indica una brecha diferencial. Tal brecha diferencial hace que la salida del controlador $u(t)$ mantenga la señal de error haya rebasado ligeramente el valor cero. En algunos casos, la brecha diferencial es el resultado de una fricción no intencional o movimiento perdido; sin embargo, a veces se provoca de forma deliberada para impedir la acción excesivamente frecuente del actuador y elemento final de control. Suponga un sistema de nivel liquido donde una electroválvula controla el liquido de entrada. Esta válvula esta abierta o cerrada. Con este controlador de dos posiciones, el liquido de entrada es, o bien una constante positiva o bien cero. Como se puede ver en la figura 14, la señal de salida fluctúa continuamente entre los dos limites requeridos para producir la acción del actuador desde una posición fija a otra. Nótese que la curva de salida sigue una de dos curvas exponenciales, una correspondiente a la curva de llenado, y la otra a la curva de vaciado. Tal oscilación entre dos limites es una respuesta típica de un sistema bajo la acción de un controlador on-off.

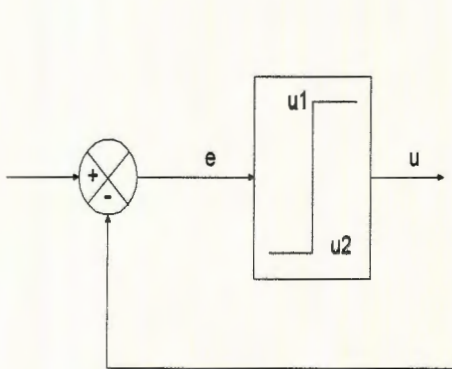


figura 13a Controlador on - off.

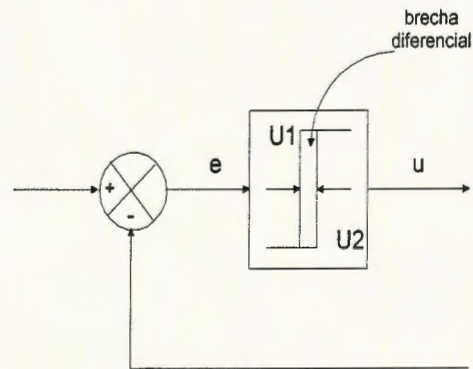


figura 13b Controlador on - off con brecha diferencial o zona muerta.

De la figura 14, se puede deducir que la amplitud de la oscilación de salida puede reducirse disminuyendo la brecha diferencial. Sin embargo, la reducción de la brecha diferencial aumenta la cantidad de conmutaciones por minuto, y por tanto reduce la vida útil de los componentes.

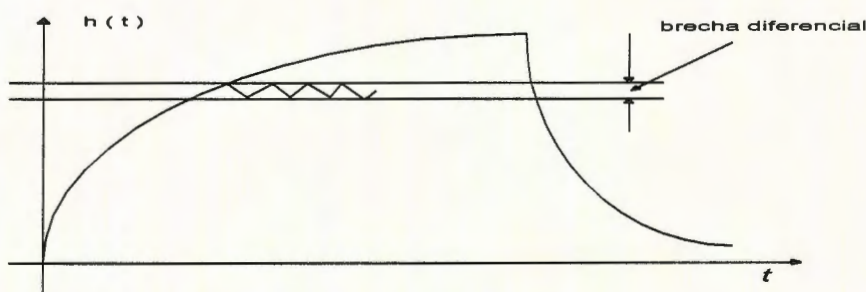


figura 14 Nivel de $h(t)$ en función de t .

5.4 CRITERIOS DE EVALUACION DE UN SISTEMA DE CONTROL

Los criterios a considerar en el diseño o en la selección de un sistema de control son muy variados, sin embargo, tres criterios básicos que deben tenerse en cuenta son:

a) Estabilidad: todo sistema de control debe ser estable es decir, no deberá presentar oscilaciones persistentes, que puedan alcanzar amplitudes extremas o que puedan provocar la destrucción del sistema. En un sistema de control estable las oscilaciones que puedan ocurrir al haber un cambio en el valor de referencia o al presentarse un cambio de carga, se debe amortiguar rápidamente.

b) Exactitud: el grado de exactitud requerido depende de la aplicación. El sistema en su funcionamiento deberá de producir una salida lo suficientemente exacta. Es decir, un sistema de control debe poder reducir a cero, o a un valor muy pequeño, los errores.

c) tiempo de respuesta: sería ideal que un sistema respondiera en forma inmediata cuando se cambia el valor de referencia, o sea que la variable controlada se ajustara a su nuevo valor, apenas se cambia dicha referencia; en la realidad esto no ocurre, siempre se presenta un retardo en la respuesta a un cambio. Sin embargo en el diseño e implementación de sistemas de control se deberá tratar de reducir el tiempo de respuesta del sistema de control a cambios en la señal de referencia.

5.5 ANALISIS Y DISEÑO DE LOS SISTEMAS DE CONTROL

Los componentes que abarcan los sistemas de control son muy diversos. Pueden ser estos de tipo electromecánicos, hidráulicos, neumáticos, electrónicos, etc. En ingeniería de control, en lugar de operar con dispositivos o componentes físicos, se reemplazan por diagramas topográficos, o por modelos matemáticos.

Obtener un modelo matemático razonablemente exacto de un componente es extremadamente difícil, y es uno de los problemas mas grandes en ingeniería de control. Un modelo de este tipo debe representar los aspectos esenciales de un componente físico. Las predicciones del comportamiento de un sistema, basadas en diagramas de bloques o esquemáticos deben ser bastantes aproximado. El uso de estos modelajes permite desarrollar una teoría de control unificada, basada en ecuaciones de tipo diferencial.

Después de obtener un modelaje de un sistema, no interesa la naturaleza de los componentes, si son neumáticos, eléctricos, etc. En la practica, los sistemas están compuestas a muchas restricciones, o no ser lineal, y en tales casos el análisis de estos sistemas se vuelve complejo; a continuación se presenta en la figura 15 un diagrama general topográfico del sistema de adquisición de datos expuesto en este trabajo.

5.6 PROBLEMAS POTENCIALES EN LOS SISTEMAS DE CONTROL

El primer problema es el overshoot (incremento exagerado), que sucede cuando

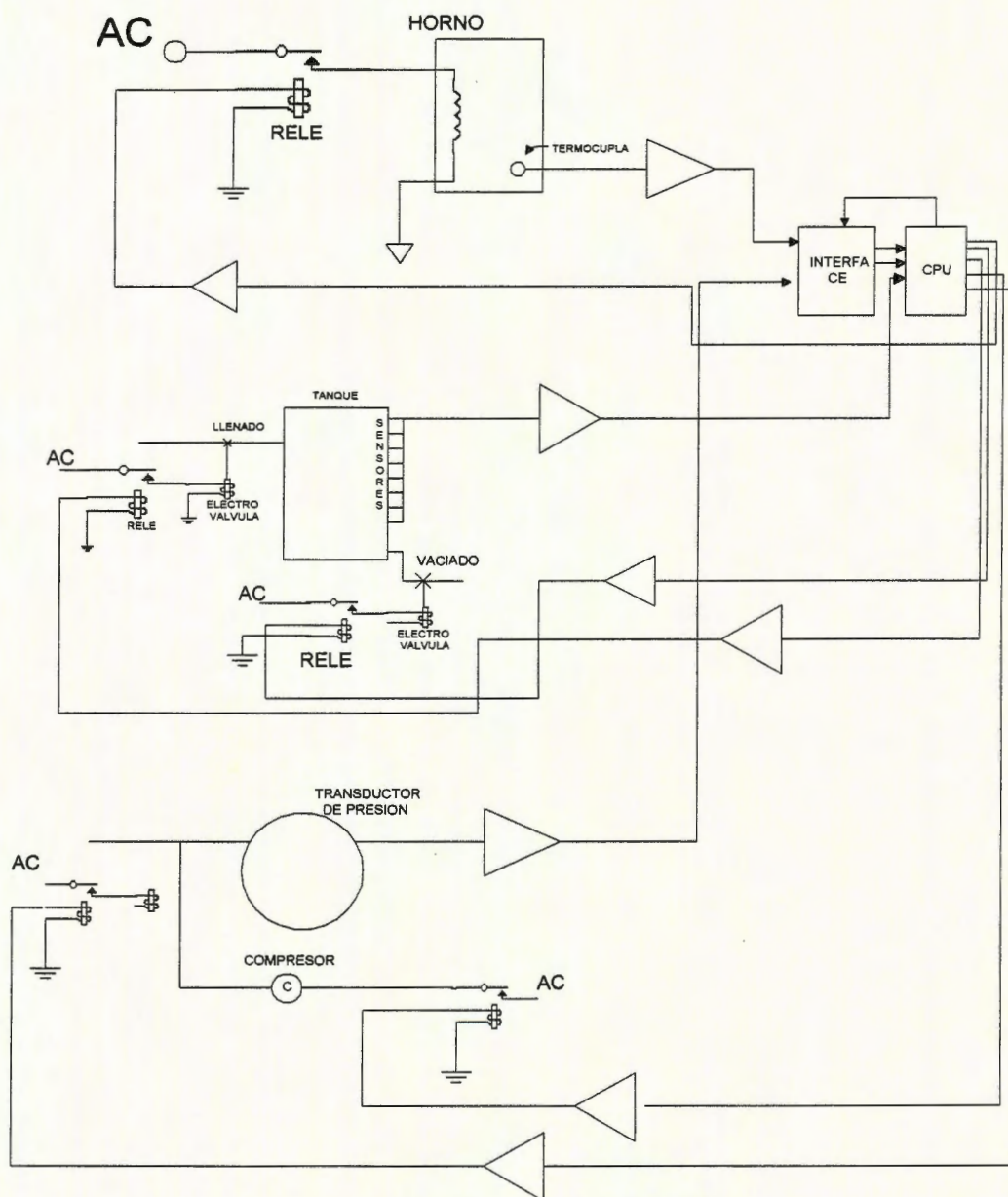


figura 15 Diagrama esquemático de un controlador

se cambia el set point, en este caso la variable cambia produciendo una oscilación rápidamente que provoca un rebote. El tiempo que toma el rebote para restablecerse la variable dentro de un rango específico produce un error llamado tiempo de restablecimiento.

Otro problema es el error residual que es producido por la respuesta del sistema de control cuando se incrementa la carga, esto se debe al tiempo de retardo que toman todos los dispositivos electrónicos conectados al sistema para hacer cambiar el estado de el controlador provocando siempre un error residual entre la variable fijada y la salida.

En sistemas de control de procesos sofisticados esto se corrige con circuitos complejos de realimentación que optimizan el cambio entre la carga o set point.

6. DESCRIPCION DEL HARDWARE DEL SISTEMA

El hardware del sistema se divide en dos etapas principales, la etapa de adquisición que es la que se encarga de leer el valor de las variables e introducirlas a los registros del computador y la etapa de control es utilizada para gobernar los diferentes actuadores basado en la comparación de los puntos de set (set points).

6.1 ETAPA DE ADQUISICIÓN

Antes de diseñar un sistema de adquisición de datos y aplicar los conceptos de muestreo se debe determinar la naturaleza de las señales a ser procesadas, es decir clasificar si las señales son dinámicas o continuas, estables o aleatorias, al igual que sus rangos de frecuencia y las características del ruido. El tipo de señales a adquirir son de tipo aleatorias y sus variaciones se encuentran en un rango de frecuencia bajos.

Según mediciones realizadas para cada variable a sensar, se determinó la frecuencia de muestreo siendo los resultados los siguientes:

La capacidad de almacenamiento del recipiente es de 32 litros, el tanque se llena a una razón de 0.13 lts./seg. La cantidad de electrodos máxima por limitantes del ADC0809 es de 8, ya que su bus de datos es de 8 bits. Un monitoreo conveniente para estas variables es de cada 4 litros, porque el caudal especificado anteriormente toma 30 segundos en ingresar un volumen de 4 litros al barril, según Nyquist se debe hacer un muestreo cada 15 segundos lo que corresponde a una frecuencia mínima de 0.06 Hertz. El ADC0809 hace un muestreo de estas variables cada 62.5 KHz. lo que es mayor que 0.06 Hz., con esto garantizamos cumplir con el criterio de Nyquist.

En el caso de la temperatura y la presión los cambio no son tan inmediatos debido a factores tales como el retardo de transporte, y a que en la mayoría de procesos en los que se

mide cualquiera de estas dos variables, el cambio no es brusco y generalmente interesa medir la ocurrencia de valores críticos (temperaturas y presiones extremadamente altas o bajas), por lo tanto de acuerdo a la aplicación particular se escoge una frecuencia de muestreo específica. Para el proceso en estudio, los cambios se registraban cada 15 segundos para la temperatura y 12 segundos para la presión (valores aproximados, ya que estos tiempos también dependen de factores ambientales externos al sistema) lo que da un rango de frecuencia de 0.066667 Hertz y 0.083333 Hertz respectivamente. La frecuencia de muestreo del ADC0809 es de 62.5 KHertz, el usuario tiene la opción de especificar frecuencias de muestreo diferentes por medio de software según sean los requerimientos del proceso en análisis. Los factores que intervinieron en la selección de los diferentes transductores fueron: facilidad de obtención en el mercado, factores económicos y compatibilidad con el sistema.

6.1.1 Circuito de adquisición

La utilidad del presente circuito es la de servir como un interfaz entre el computador y las diferentes señales a medir. En el anexo A8 se muestra el circuito .

La adquisición de datos inicia por medio de los distintos transductores, los cuales convierten una variable física a una señal eléctrica que es alimentada al ADC0809 (Convertidor A/D) y adquirida por el computador a través de un PPI 8255A.

Los datos analógicos a medir son presión y temperatura. Para la adquisición del dato de temperatura se utiliza una termocupla tipo J, que es conectada a una red de amplificadores operacionales que funcionan en las configuraciones de seguidor de voltaje, diferenciador y amplificador no inversor (ver apéndice A9), con el fin de asegurar un voltaje de aproximadamente cero voltios para la temperatura menor y cinco voltios para la temperatura máxima a medir (600 °C).

La adquisición del dato de presión utiliza una configuración similar, es decir que el sensor de presión se conecta a una red de amplificadores operacionales para asegurar el

voltaje de cero cuando la presión es mínima y cinco voltios cuando es máxima (16 psi). La adquisición del dato de nivel se hace valiéndose de la propiedad conductora del agua; se asigna un voltaje de referencia común y a medida que el agua aumenta o disminuye su nivel, este voltaje es transmitido a los diferentes electrodos, para posteriormente activar las entradas del buffer 74LS244 obteniéndose de esta forma un código que representa el nivel alcanzado por el líquido, y este código es introducido al computador.

El convertidor ADC0809 utilizado es un sistema de adquisición de datos de ocho entradas de las cuales se ocupa la entrada uno para temperatura, la entrada dos para presión y las seis restantes para futuras expansiones. El reloj del ADC 0809 es generado por un circuito RC en conjunto con los inversores del 74LS04 para obtener una frecuencia de 500 kHz. La lógica de control del ADC0809 es proveída por el computador a través del puerto C del PPI8255A. Los datos de temperatura y presión son multiplexados con los datos adquiridos en el sensor de nivel para poder ser introducidos al computador a través del puerto A del 8255A, dicho multiplexeo se logra por medio de los Buffers de tres estados 74LS241 de acuerdo con la siguiente tabla de verdad:

DIRECCIONES		VARIABLES
0	0	TEMPERATURA
0	1	PRESIÓN
1	0	N/A
1	1	NIVEL

Tabla 6. Direccionamiento de las variables.

Los datos de direcciones son proveídos por el computador a través de la mitad del puerto C del PPI 8255A. Para las direcciones del 00 al 10 (binario), el ADC0809 multiplexea los canales analógicos y en el enable del 74LS241 se encuentra un cero lógico

el cual deja pasar los datos del ADC, cuando se llega a la dirección 11 (binario), la compuerta AND del 74LS08 manda un uno lógico al enable del 74LS241 desactivando así la entrada del ADC0809 al computador y activando la entrada del 74LS244 (detector de nivel) al computador .

El 74LS138, se utiliza como decodificador de direcciones para el PPI8255A, fijando las direcciones siguientes para los puertos:

DIRECCIÓN	FUNCION
260H (608 DEC)	PUERTO A
261H (609 DEC)	PUERTO B
262H (610 DEC)	PUERTO C
263H (611 DEC)	PALABRA DE COMANDO

Tabla 7. Direccionamiento para registro del PPI.

La palabra de comando que se escribe en la dirección 611 (DEC) de la PPI8255A es 152 (DEC), la cual fija la parte alta del puerto C de la PPI8255A como entrada y la parte baja como salida; el puerto B como salida y el puerto A como entrada, esto se debe a que mediante el dato 152 la PPI queda fijada en modo cero. La parte baja del puerto C maneja las direcciones A y B, el start conversión, el address load control y el out enable del ADC0809; y en la parte alta se lee el dato que viene del pin de end of conversión del ADC para mantener el control .

A través de los diferentes puertos del PPI, se introducen los datos, los cuales son un reflejo de las condiciones físicas del proceso. Estos son obtenidos por medio de los diferentes transductores instalados en el sistema, (presión, nivel, temperatura). Los de presión y temperatura, son de fácil comprensión, ya que su salida es proporcional a la variable medida.

Para poder muestrear la salida de los diferentes transductores, las señales deben de ser proporcionalmente amplificadas, el valor de la proporción debe de ser conocido, para poder obtener una lectura real de la variable en prueba (medición), esta amplificación proporcional se logra por medio de amplificadores operacionales y a continuación se presentan los cálculos para determinar el valor de la proporción.

Como voltaje máximo se asume 5 Volt (Voltaje Máximo de entrada al ADC). Como voltaje mínimo se asume 0 Volt. Con estos datos y con una configuración del amplificador operacional como amplificador no inversor se utiliza la siguiente formula:

$$AV = 1 + Rf/Ri \quad AVf = Vo/Vi$$

$$\text{De donde } Vo \text{ máx} = 5 \text{ V}$$

$$Vi \text{ max a } T^{\circ}C = 42.922 \text{ mV}$$

$$Vo \text{ max} = 5V$$

$$AV = 5V/42.922mA = 116.49$$

$$\text{Si } Rf = 220 \text{ K}\Omega; AV = 1 + Rf/Ri$$

$$116.49 = 1 + 220 \text{ k}\Omega/Ri$$

$$Ri = 220 \text{ k}\Omega/116.49$$

$$Ri = 1904.92 \Omega$$

$$Ri \text{ valor comercial} = 2 \text{ k}\Omega$$

Ocasionalmente, surge la situación donde una entrada TTL debe conservarse normalmente bajo (low) y luego mandarla a alto (high) por la acción de un interruptor mecánico o a través de algún contacto físico (para el caso a través del agua), para lograr este objetivo se utiliza una configuración, como la mostrada en la figura 16.

La resistencia sirve para asegurar un bajo mientras el interruptor se halla abierto. Para cumplir con esta condición, R deberá mantenerse lo suficientemente pequeño de manera que el voltaje generado a través de él por la corriente I (IL) que fluya hacia afuera de la entrada del circuito integrado conectada a tierra no exceda V(IL) máximo.

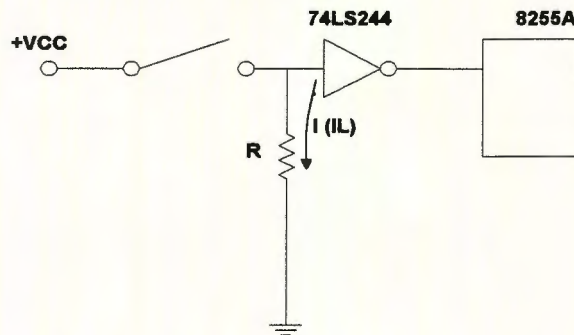


Figura 16. Detector de nivel

De este modo el valor de R esta dado por:

$$I(IL) \times R_{max} = V(IL)_{max}$$

R deberá conservarse debajo de este valor para asegurar que la entrada del integrado estará en un nivel bajo aceptable en tanto que el interruptor este abierto. El valor mínimo de R se determina por la corriente obtenida de la fuente de 5 V cuando el interruptor esté cerrado. En la práctica, esta corriente deberá ser minimizada conservando a R ligeramente debajo de R máximo.

Para el caso particular, se usará el IC 74LS244, de donde se sabe que:

$$V(IL)_{max} = 0.8 \text{ voltios}, I(IL) = 0.2 \text{ mA.}$$

Aplicando la ecuación anterior:

$$R_{max} = 0.8 / 0.2 \text{ mA} = 4000 \Omega.$$

Un valor comercial de esta resistencia, y que no exceda el valor calculado es 3900Ω.

El sistema donde se implementara la adquisición de datos es un barril en el cual se introducirá agua y se sensorá el nivel que ésta alcance dentro del mismo. Este barril tendrá un conducto de entrada en la parte superior y uno de salida en la parte inferior para que sea controlado el flujo de agua y así el nivel mediante los actuadores para el caso electrovalvulas proporcionales. También se sensorá la temperatura aisladamente, por medio de una termocupla. La temperatura se hará incrementar o disminuir con unas hornillas, los

actuadores para este caso se obtendrán de circuitos con relés que interrumpirán la entrada de voltaje que alimenta las resistencias de las hornillas. La medida de presión se hará también aislada, mediante un sensor del tipo integrado. La variación de la presión se controlara mediante una válvula de escape para que la presión no sea demasiado alta.

6.2 ETAPA DE CONTROL

A continuación se describe el funcionamiento de la etapa de control del sistema de adquisición de datos. El cual involucra lo que es el manejo de las variables a controlar mediante actuadores, los cuales son manejados por medio del software para llevar a cabo el control.

El tipo de control de procesos que se utiliza es un sistema ON-OFF para el caso de la temperatura y la presión el cual se encarga de proveer una señal de encendido o apagado a los diferentes actuadores del sistema cuando una variable esta en el punto de set point o fuera de este. En el caso del nivel se desarrolla un control proporcional. El controlador es implementado en tres procesos aislados: temperatura, presión y nivel tal como se describe a continuación.

6.2.1 Control de Temperatura

Una termocupla es colocada dentro de un horno con el propósito de mantener condiciones de temperatura estables que conservan el calor. La temperatura dentro del horno debe ser mantenida dentro de un rango prefijado por el usuario. Inicialmente, el sistema compara desde la primer muestra tomada y determina si la temperatura esta por debajo del limite inferior; si la comparación resulta verdadera una resistencia de 500 Watts es conectada por medio de un relé para poder alcanzar el nivel. Cuando se alcanza el límite superior de temperatura, se desconecta la resistencia para mantener la temperatura deseada.

6.2.2 Control de presión

Para este proceso, se coloca el sensor de presión SEMSYN SPX100A en un neumático, el neumático es presurizado por medio de un compresor que provee una presión de hasta 200 PSI en la válvula del neumático se coloca un conector tipo T que tiene la función de conectar el compresor en un extremo de la T y una electroválvula de purgado para liberar presión. El usuario fija un valor de presión a ser mantenido, este dato es comparado con el primer dato adquirido, si el dato adquirido es menor que el dato prefijado, se activa un relé que le da paso al voltaje de alimentación del compresor. Cuando se alcanza el nivel de presión deseado, se desactiva el relé y se abre el contacto que se encarga de proveer el voltaje al compresor. Si el dato adquirido es mayor que el dato prefijado, se activa un relé que conmuta un contacto normalmente abierto a la posición de cerrado y deja pasar el voltaje de alimentación para una electroválvula que deja salir el aire comprimido para liberar la presión dentro del neumático.

6.2.3 Control de nivel

El proceso de control de nivel es implementado en un tanque con capacidad para 32 Lts con intervalos de medición cada 4 Lts. El usuario prefija un valor de nivel máximo a ser controlado este valor es comparado con el dato adquirido, es decir, si el dato adquirido es menor, el software genera una palabra digital de cuatro bits para ser introducida al convertidor digital a análogo DAC555JN, el cual genera un voltaje proporcional a la palabra convertida. A medida que el tanque se está llenando, la palabra a convertir tiene un valor menor y por lo tanto el voltaje de salida es menor, lo cual ocasionaría un cierre proporcional de una electroválvula que se conectara a la salida.

Las señales que controlan el hardware son generadas por medio del software donde se efectúa una comparación entre las señales obtenidas y las variables que el usuario desea fijar. Los resultados de esta comparación son enviados al puerto B de salida del PPI8255A, por medio de compuertas inversoras, se obtiene el complemento de la palabra de control

para poder manejar unos transistores 2N5087 tipo PNP que a su vez manejan unos relés de tipo DPST (double pole single throw) que se encargan de conmutar el voltaje de alimentación para los diferentes actuadores. Para efectos de amplificación de la señal de salida de la PPI8255A, se utiliza por conveniencia una compuerta AND como buffer. Los cálculos de la resistencia de polarización del transistor se presentan a continuación y el diagrama del circuito de manejo (driver) se muestra en la figura 17.

$$I_c = \beta I_b$$

$$0.006 = 180 I_b$$

$$I_b = 0.006/180$$

$$I_b = 3.3 \text{ E-5 Amp.}$$

$$5 \text{ V} - I_b R_b - 0.7 = 0$$

$$R_b = 4.3 / 3.3 \text{ E-5}$$

$$R_b = 120 \text{ KOHM}$$

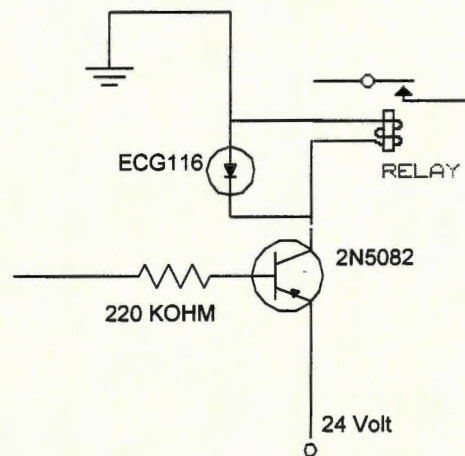


figura 17 Diagrama del circuito manejador de relé

7. SOFTWARE

El software es la secuencia lógica de ordenes que se le dan al computador para crear un programa. Para el caso el software a utilizar es Visual Basic, que se encarga de manejar la adquisición de datos y procesar la información. A continuación se explica su funcionamiento, y su ambiente de trabajo.

7.1 SISTEMA OPERATIVO WINDOWS

Las interfases gráficas del usuario, conocidas como GUI's (graphics user interface), han revolucionado la industria de la computación. El uso de interfases gráficas por medio de cajas de dialogo, ejecución de comandos por selección y presentación gráfica, hacen que una sesión en la computadora se vuelva fácil y libre de temores ya que el usuario no tiene que perder tiempo en recordar largos enunciados de comandos para una situación en particular (ejemplo: darle formato a un diskette), recordar la vía de acceso para un archivo en un directorio de 7 ramas o subdirectorios. Existen diferentes sistemas operativos que hacen uso de los GUI's, sin embargo el preferido y más utilizado es el Microsoft Windows. Cuyo nombre proviene del hecho de que el ambiente de trabajo se asemeja a una ventana o área rectangular en la cual se encuentran unos elementos gráficos conocidos como iconos (ver figura 18). Las ventanas se encuentran dispuestas en un área conocida como la carpeta o desktop, en la cual se pueden reordenar las ventanas de la forma que el usuario lo prefiera. Las ventanas se pueden abrir, cerrar, maximizar, minimizar, mover, crear, y borrar.

Cada ventana representa un grupo que tiene iconos o elementos que al efectuar una acción de doble click con el mouse, se ejecuta un comando. El objetivo de ordenar los

iconos en grupos, es para que el usuario tenga una idea de que es lo que se ejecutará al activar tal o cual icono (ejemplo: grupo games, grupo Lotus Applications, etc.).

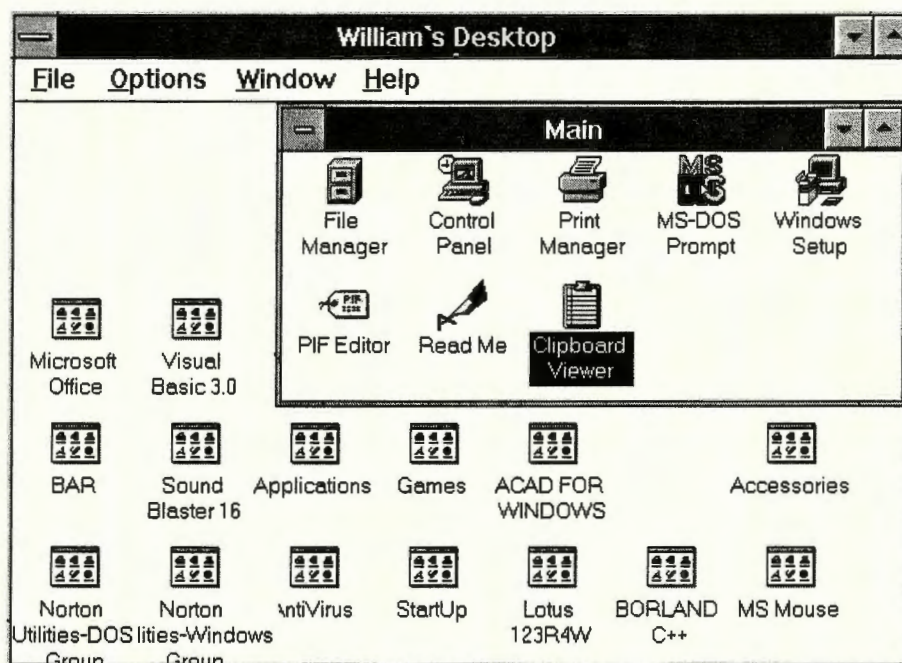


Figura 18. Ambiente Windows

7.2 VENTAJAS DEL SISTEMA OPERATIVO WINDOWS

Las ventajas que ofrece el sistema operativo Windows respecto al sistema operativo convencional DOS aparte de poseer una interfase amigable para el usuario son:

- La habilidad de multitarea.
- Objetos manejados por eventos.
- Transferencia de información a través del clipboard o portapapel.
- Uso de object linking and embedding (objetos incrustados) OLE.
- Uso de dynamic data exchange (intercambio dinámico de datos) DDE.

- Configuración de recursos para Windows en general.

7.2.1 Multitarea

La característica de multitarea permite poder correr más de un programa al mismo tiempo. La multitarea resuelve una serie de problemas tales como: ¿Qué programa obtiene el acceso a la pantalla? ¿Cómo es compartida la memoria? ¿Si más de un programa esta utilizando la pantalla, cual programa es el que recibe datos del teclado? ¿Cuál programa responde a los movimientos del mouse? ¿Qué si dos programas desean utilizar el puerto serial a la vez?

7.2.2 Objetos Manejados por eventos

La mayoría de los programas para DOS están escritos como lazos (loops) o bifurcaciones que realizan operaciones de entrada/salida cuando se necesita. El programador decide cuando es tiempo de chequear si hay entrada del teclado o del mouse. Los programas para Windows pueden recibir información de diferentes periféricos a cualquier momento sin necesidad de utilizar lazos que hacen uso del sistema y previenen que otras aplicaciones corran.

7.2.3 El Clipboard o Portapapeles

El portapapeles sirve de espacio de almacenamiento temporal para aquella información que se desea compartir. Es posible copiar o cortar información procedente de una aplicación al portapapeles y posteriormente transferir dicha información desde el portapapeles a otras aplicaciones.

7.2.4 Uso del OLE

En términos simples OLE es una forma de compartir datos entre programas, sin necesidad de utilizar los recursos que ofrece el clipboard para cortar un objeto de una

aplicación y luego pegarlo en otra. OLE provee una integración entre aplicaciones de tal forma que los usuarios pueden insertar por ejemplo una hoja electrónica de Excel en una aplicación de Visual Basic. Cuando se ejecute la acción click en la hoja electrónica, la aplicación de Excel corre en el fondo (background). De esta forma los usuarios no tienen que abandonar una aplicación para consultar o actualizar datos, figuras o gráficos de otra aplicación.

7.2.5 DDE

DDE ofrece un método para compartir información entre dos aplicaciones, de la misma forma como funciona un OLE con la diferencia de que la transferencia de información se da en un solo sentido que va de la aplicación que origina los datos (servidor) a cualquier otra aplicación Windows que soporte esta transferencia de información. Técnicamente una conversación DDE es entre dos ventanas, y para Windows resulta que la mayoría de controles son ventanas así como formularios también.

7.2.6 Configuración de recursos para Windows en general

Esto significa que las configuraciones de teclado, impresor, mouse y tarjeta de vídeo funcionan para todos los programas que corren en ambiente Windows, a diferencia del ambiente DOS en el cual se debe configurar cada uno de los dispositivos mencionados anteriormente.



7.3 LENGUAJE DE PROGRAMACION VISUAL BASIC

Visual Basic es un lenguaje de programación, que se adecua a una gran cantidad de aplicaciones. Visual Basic esta estrechamente relacionado con windows, de hecho Visual Basic es un lenguaje de programación para ambiente windows. El desarrollo de las aplicaciones para windows, era una tarea compleja y mucho más difícil de desarrollar que

una aplicación para DOS. Los programadores tenían que considerar muchas variables tales como la posición y el manejo del mouse, si el usuario estaba haciendo un click normal o un doble click, etc. Desarrollar una aplicación en windows requería de expertos programadores en lenguaje C y ensamblador. Visual Basic cambió por completo esta situación. Aplicaciones sofisticadas para windows pueden ser desarrolladas en una fracción de tiempo del que se requería anteriormente.

7.4 CARACTERISTICAS DE VISUAL BASIC

Visual Basic permite agregar menús, cajas de textos, botones de comando, botones de opciones (para hacer selecciones exclusivas), chequear cajas para selecciones no exclusivas, cajas de listados, barras para examinar. Se pueden crear rejillas para el manejo de datos tabulados, comunicar con otras ventanas de windows y acceder bases de datos. Se pueden tener múltiples ventanas en un ventana. Estas ventanas tiene acceso completo al portapapeles (clipboard) y a la información de la mayoría de las otras ventanas que se encuentran corriendo simultáneamente. Visual Basic permite la construcción fácil de programas largos a través de técnicas de programación modular. Esto significa que se puede detener un programa y depurarlo fácilmente, y por lo tanto disminuye el riesgo de error; posee además un interpretador inteligente, el cual detecta, e incluso sugiere los cambios necesarios para corregir una rutina de programa, posee una extensiva ayuda (help) en línea para rápida referencia mientras se desarrolla una aplicación.

7.5 REQUERIMIENTOS DE HARDWARE PARA EJECUTAR VISUAL BASIC

Visual Basic es un programa sofisticado. La edición estándar requiere una computadora IBM compatible y que utilice al menos un procesador 386. Para el sistema completo de Visual Basic se requiere de:

Un disco duro con 13 Mega bytes libres.

Un mouse o algún otro dispositivo puntero compatible con el mouse.

Un monitor de gráficos y tarjeta con resolución EGA (con mayor resolución es mejor).

Versión de MS o PC DOS 3.1 o mayor.

Versión de Microsoft Windows 3.0 o mayor, ejecutándose en modo estándar o mejorado (preferible versión 3.1).

Al menos 2 MB de RAM (de preferencia 4 MB).

7.6 EL AMBIENTE DE VISUAL BASIC

Con el fin de conocer sobre Visual Basic se presenta posteriormente el uso de menús y ventanas que lo componen; se explicará brevemente la ayuda en línea (on line help), las herramientas de edición (editing tool), y el manejo para construcción de archivos.

Los valores que Visual Basic fija por defecto en la construcción de un proceso de diseño de una aplicación, pueden ser cambiados con facilidad. Las ventanas construidas por defecto son del tamaño adecuado, pero estas pueden ser movidas y cambiar su tamaño como el usuario lo desee. Por supuesto Visual Basic permite cambiar todos los valores, por defecto, pero no es buena idea hacer cambios al azar.

7.6.1 La pantalla principal (Main Screen)

Cuando se inicializa Visual Basic, se presenta la pantalla de derechos de copia (copyright screen) indicando quien posee la licencia para copia de este, después de un breve retardo automáticamente inicializa Visual Basic. Visual Basic recuerda como se arreglo la pantalla la ultima vez, debido a eso la pantalla podría no siempre lucir igual o lucir diferente de la figura que se muestra.

La pantalla ciertamente luce saturada (llena), se pueden observar cinco partes en la pantalla principal de Visual Basic, la ventana de propiedades (property window), la ventana de proyecto (project), la ventana del form, la caja de herramientas (tool box), y la barra de menús (ver figura 19). La ventana de propiedades es utilizada para personalizar un form o un control. La otra ventana llamada la ventana de proyecto (project window), se utiliza para ordenar la secuencia de los archivos, módulos y forms que componen un programa o proyecto. Se presenta a continuación una descripción de las otras partes de la pantalla principal.

7.6.2 Título (title bar)

El título es una barra horizontal localizada en la parte superior de la pantalla, en ella esta escrito el nombre de la aplicación y es común para todas las aplicaciones de Microsoft Windows. La interacción entre el título y el usuario es manejada por Windows y no por la aplicación. Bajo la barra de títulos y la barra de menú se encuentra el área de trabajo (form). La aplicación a diseñar es responsable totalmente de la apariencia, y de la respuesta de los objetos que se colocan en esta área.

7.6.3 Barra de menús (menús bar)

Una de las características que hace el uso de Windows fácil e interactivo es el uso de los menús tipo pull down. En Visual Basic la barra de menú permite obtener todos los comandos o herramientas para desarrollar, probar y almacenar una aplicación. El menú de File contiene los comandos para trabajar con los archivos que componen una aplicación, el menú de Edit contiene muchas de las herramientas de edición útiles para escribir el código que activa la interfase que se diseña para una aplicación, el menú de View provee acceso rápido a las diferentes partes del programa, el menú Run permite probar la aplicación

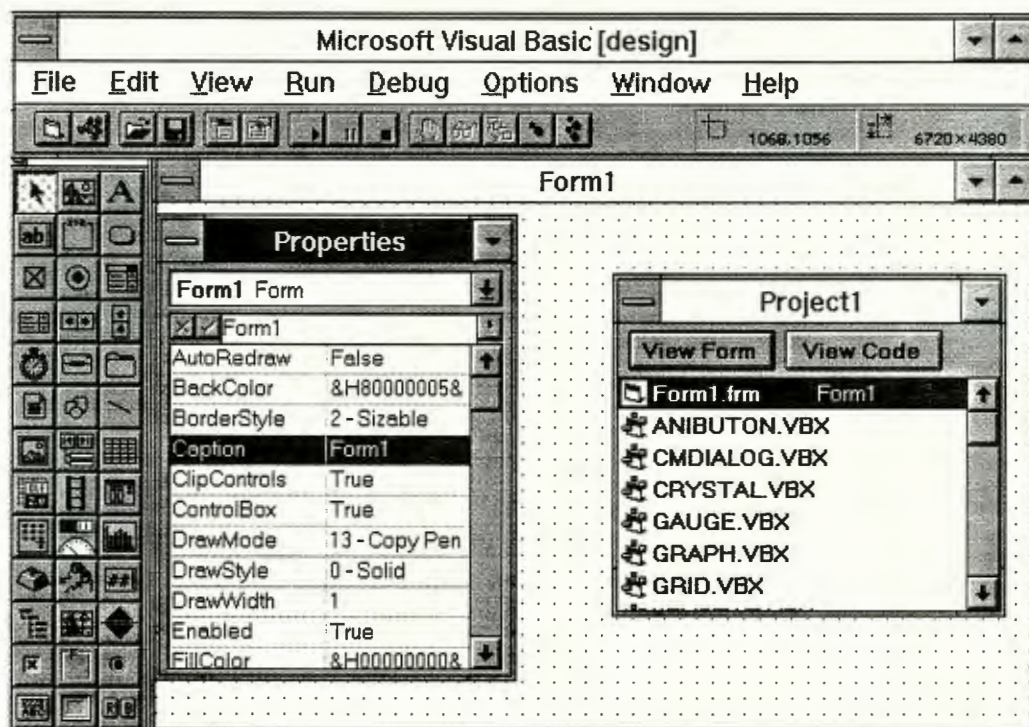


Figura 19. pantalla principal de Visual Basic

mientras se esta desarrollando. El menú Debug permite acceder las herramientas para corregir los problemas (depurar). El menú de Options permite controlar el ambiente Visual Basic, el menú window permite un acceso rápido a las diferentes ventanas que constituyen el ambiente Visual Basic. Este menú permite diseñar menús de pull down para diferentes aplicaciones. Finalmente el menú Help permite el acceso al sistema de ayuda detallado en Visual Basic.

7.7 LOS ELEMENTOS DE VISUAL BASIC

Para tener una visión clara de la forma en que Visual Basic trabaja es necesario describir con claridad los elementos que intervienen para poder desarrollar una aplicación. Los elementos que interactúan en una aplicación son: la ventana de proyectos, los forms,

objetos, las propiedades, los eventos, las subrutinas, los módulos, la ventana de códigos, y la caja de herramientas.

7.7.1 La ventana de proyectos (project window)

Esta muestra la lista de todos los forms nombrados con una extensión frm. Visual Basic almacena cada form como un archivo separado y usa el archivo con extensión Mak para guardar las pistas de donde ellos son almacenados para que de esta forma la aplicación los llame cuando se ejecuten.

7.7.2 Los forms

Son la unidad primaria de organización en Visual Basic y básicamente es una ventana que el usuario personaliza, en la cual se insertan todos los objetos que interactuarán en la aplicación.

7.7.3 Los objetos

Los objetos son una serie de elementos predefinidos que responden cuando ocurre un evento, donde cada objeto tiene asociado una serie de propiedades y métodos (subrutinas o funciones que operan en un objeto).

7.7.4 Ventana de Propiedades

Es una ventana en la cual se muestran las cualidades o atributos que definen las características de cada objeto (tal como el tamaño, color, localización en la pantalla) o sus conductas (por ejemplo cuando esta habilitado o no).

7.7.5 Los eventos

Son todas las acciones que podrían sucederle a un objeto cuando el usuario ejecuta el programa. Dentro de los ellos se declaran las instrucciones a las cuales responderá el objeto.

Esto se lleva acabo por medio de una codificación.

7.7.6 Las Subrutinas

Una subrutina es como un pequeño programa dentro de un programa grande que realiza una tarea limitada y que se encuentra disponible para ser llamada a ejecutar esa tarea desde cualquier punto del programa. Usualmente las subrutinas son utilizadas para ahorrarse de repetir una y otra vez las mismas instrucciones dentro del programa.

7.7.7 Los módulos

Un modulo es como una subrutina con la diferencia de que los módulos pueden ser accesados por el programa entero (varios forms). Un modulo es similar a un form con la diferencia de que los módulos no son visibles y no tienen eventos asociados a ellos. El propósito de un modulo es contener subrutinas o funciones y declarar variables que están disponibles para todo el programa.

7.7.8 La ventana de códigos

Es activada cada vez que se hace un doble click en un form o en un control, también se puede acceder desde la ventana de proyecto o presionando la tecla de función 7 (F7), la ventana de código cuenta con un área para la edición o introducción de comandos o códigos y con dos cajas de listados (list box). La caja de listado izquierda, llamada object box, lista todos los controles en el form, más un objeto llamado general que contiene un código común que puede ser utilizado por todos los procedimientos que se presentan en el form. La caja de listado derecho llamado proc, es la caja de listado de procedimiento, esta caja presenta todos los objetos reconocidos por el objeto seleccionado en la caja de list object.

7.7.9 La caja de herramientas

Esta contiene los objetos de Visual Basic (controles) que son insertados en un form

para responder a un evento. Cada objeto tiene propiedades que pueden ser cambiadas en cualquier momento fuera del programa o durante la ejecución del programa. Para insertar un objeto en el form basta con hacer doble click sobre él.

7.8 COMO DESARROLLAR UNA APLICACION EN VISUAL BASIC

El primer paso para desarrollar una aplicación en Visual Basic es determinar lo que el usuario verá. En otras palabras, diseñar las pantallas, cuales menús se necesitan, que tan largas serán las ventanas que la aplicación ocupará, definir si el usuario podrá cambiar el tamaño de las ventanas, en donde se colocaran los botones de comando, es decir los botones en los cuales el usuario ejecutara un evento para activar las aplicaciones. En Visual Basic los objetos que un programador coloca en una ventana son llamados controles.

Lo que hace que Visual Basic sea diferente a otro lenguaje de programación es la facilidad con que se pueden diseñar las pantallas; literalmente se pueden dibujar las interfaces visuales, los botones de comandos, las cajas de texto y los otros controles que se utilizarán en la aplicación y que responderán automáticamente a las acciones del usuario tales como movimientos y acciones de click con el mouse. También Visual Basic posee la ventaja para crear menús rápidamente.

Luego de diseñar la interface visual, esta responderá a cualquier función como ocurre en los programas tradicionales. Cada objeto en Visual Basic reconocerá el evento que se le halla asignado (un click, doble click, o alguna tecla) y responderá a el dependiendo del código escrito. Siempre es necesario escribir una codificación para hacer que el control responda al evento. Esta es la diferencia fundamental de Visual Basic respecto a la programación convencional.

Los programación convencional es estructurada, es decir todos los programas corren de arriba hacia abajo comenzando de la primera línea y moviéndose con el flujo del programa a las diferentes partes que sean necesarias. Visual Basic trabaja completamente

diferente. El núcleo de un programa en Visual Basic es un conjunto de piezas independientes que por medio de códigos que son activados responden únicamente a los eventos para los cuales están programados a reconocer, esto es un cambio fundamental. Ahora en lugar de diseñar un programa que haga lo que el programador piense que debería suceder, el usuario es el que tiene el control.

La mayoría de los códigos de programación en Visual Basic le dicen al programa como responder cuando los eventos ocurren, esta acción es conocida como event procedures. Esencialmente todo objeto que se ejecuta en un programa de Visual Basic, es ya sea un event procedure o usado por un event procedure que le ayuda a un procedimiento a ejecutar su tarea.

De hecho para establecer que Visual Basic es fundamentalmente diferente de los lenguajes de programación ordinarios, la documentación utiliza el término proyecto en lugar de programa, para referirse a la combinación de código de programación con la interface del usuario para poder crear una aplicación en Visual Basic.

Los pasos a seguir para diseñar una aplicación en Visual Basic son los siguientes:

1. Diseñar las ventanas que el usuario verá
2. Decidir a que eventos los controles en la ventana deben responder
3. Escribir los códigos a los cuales responderán estos eventos
4. Escribir cualquier procedimiento que el programa necesite para que realice su función.

Al diseñar una aplicación en Visual Basic esta corre ejecutando los siguientes pasos:

1. Visual Basic monitorea las ventanas y los controles en cada una de ellas para todos los eventos que cada control puede reconocer.
2. Cuando Visual Basic detecta un evento, examina la aplicación para chequear si esta escrita una codificación para ese evento
3. Si se ha escrito una codificación, Visual Basic lo ejecuta y regresa al paso 1.

4. Si no se ha escrito una codificación para el evento, Visual Basic espera el próximo evento y regresa al paso 1.

5. Estos pasos se ejecutan cíclicamente hasta que la aplicación termina.

La clave para la programación en Visual Basic es reconocer que el programa generalmente procesa códigos únicamente en respuestas a eventos. De hecho, a diferencia de la mayoría de lenguajes de programación las líneas de ejecución en un programa Visual Basic deben estar dentro de procedimientos o funciones. Líneas de ejecución aisladas no son consideradas.

7.9 COMANDOS DE VISUAL BASIC

Cuando se introducen comandos en Visual Basic, el programa detecta si la sintaxis esta correcta, de no ser así, inmediatamente después de presionar la tecla de enter se muestra un mensaje alertando algún tipo de mala escritura o parámetros inválidos del comando, en este punto existe ayuda disponible al presionar la tecla de función 1 (F1).

Los comandos en Visual Basic rara vez utilizan número de líneas, y generalmente se escribe cada comando en su propia línea. Se puede combinar varios comandos en una sola línea colocando dos puntos (:) entre dos comandos, el número máximo de caracteres por línea es de 255. Las variables en Visual Basic almacenan información (valores). Cada vez que se utiliza una variable, Visual Basic fija un área en la memoria de la computadora para fijar una información. El nombre de una variable pueden contener hasta 40 caracteres, con la modalidad que el primer carácter debe ser una letra y los siguientes caracteres pueden ser una combinación alfanumérica. Existen ciertas palabras que no pueden ser utilizadas como variables ya que están utilizadas por Visual Basic para palabras de control.

Se describen a continuación los comandos, funciones, propiedades y eventos utilizados en la codificación del programa haciendo mención únicamente a los comandos de mayor relevancia.

Agintp, Agoutp: Estas funciones permiten el acceso de los puertos de entrada / salida

bajo windows, típicamente estas funciones son utilizadas para acceder interfaces especializadas para las cuales no existe un manejador windows (driver). La función `agintp` es usada para introducir datos de un puerto de entrada/salida. El parámetro que utiliza esta función, es la identificación del puerto (dirección) como un valor entero, el valor de retorno es el byte leído desde el puerto. La función `agoutp` es utilizada para enviar datos a un puerto de entrada/salida. Los parámetros que utiliza esta función son: la identificación del puerto (dirección) como un valor entero y el byte que se desea escribir en el puerto entrada/salida.

Declare: El declare statement es utilizado para importar una función DLL, dentro de Visual Basic. Este comando informa a Visual Basic del lugar de donde la función DLL puede ser encontrada, y además sirve para informarle al programa los parámetros que la función DLL espera y retorna. Una vez la función DLL ha sido declarada correctamente, esta funciona como si fuera una subrutina que el programador puede llamar cada vez que la necesita. La sintaxis del comando Declare para subrutina que no retorna un valor es:

Declare sub globalname lib libnames\$ [Alias aliasname\$] = [(argument list)]. Para funciones que retorna un valor es: Declare function globalname Lib libname\$ [Alias aliasname\$] = [(argument list)].

Remark (rem): son colocados en los programas para explicar las funciones del código. Los comandos `rem` no son procesados y no ocupan espacios en el código compilado, existen dos formas de indicar el comando `remark`, una forma es anteponiendo la palabra `rem` al comentario, la otra es precediendo al comentario por un apóstrofe.

End: cuando Visual Basic procesa el comando `end` el programa se detiene, todas las ventanas son cerradas y el programa es retirado de la memoria.

Caption (propiedad): Esta propiedad puede determinar el texto que aparecerá en la barra de título de un form, el título de un botón de control o el nombre de una etiqueta. Esta propiedad puede ser cambiada mientras el programa está corriendo.

Data (control): por medio de este control se obtiene acceso a las bases de datos y algún lenguaje especializado con el cual se puede obtener una base de datos. Las bases de

datos son colección de información ordenadas en filas (registros) y columnas (campos), esta información puede ser editada, ordenada de una forma especial, además se puede borrar o añadir información. Cuando se abre una base de datos el objeto Recordset es creado automáticamente para relacionar la base de datos con el programa.

Label: Un label informa al usuario el significado de algo visible en una ventana, los labels aparecen como pintados en un form y usualmente no cambian mientras el programa se ejecuta, sus usos principales son: imprimir información en un form con mayor flexibilidad que si se usara el comando print, cada comando print esta limitado a un simple fontname, fontsize u otra cualidad de cada texto. Cada label puede tener diferentes cualidades tipografías, agregando variedad a la apariencia de un form, de hecho es más fácil diseñar la apariencia de un form con labels que con print; si se moviese una caja de figura, se puede desplazar su label asociado y colocarlo de la mejor forma, mientras que no se puede apreciar la localización de un texto que se haya ocupado print hasta que el programa se ejecute.

Select Case: select case es similar a la estructura if...then, utilizada en Quick Basic, es ocupado usualmente para selecciones múltiples, es decir cuando existen dos o mas posibles selecciones. Puede trabajar con una lista posible de respuestas, diferentes para cada selección.

Show: este comando hace a un form o ventana invisible; los objetos no pueden normalmente ser creados o editados cuando su form es invisible, el usuario nunca podrá ver esos objetos al menos que sea forzado, suprimiendo el comando show del programa previa ejecución de este.

Sendkey: Este comando permite enviar combinaciones de teclas a otra programa de Windows, como si el usuario estuviese digitando estas teclas. Esta técnica permite la comunicación entre todas las ventanas de Visual Basic o con ventanas que no son de Visual Basic.

Setfocus: setfocus permite que el programa, mientras se ejecute, active un control o

form diferente. En windows, el usuario activa una lista de archivos o algún control diferente a través del mouse (mandando un click), en Visual Basic el programa puede activar estos mismos controles. Solamente un control puede ser activado a la vez, al igual que en windows solo una ventana puede ser activada (para poder recibir datos) a la vez.

Este comando proporciona mayor inteligencia al programa, y además permite dar un orden de activación a los controles, el cual es creado por el programa, con lo que se logra independizar este orden de activación de la tecla de TAB, lo que muchas veces activa controles innecesarios antes de activarse el deseado, todo esto resulta en un ahorro de tiempo.

Appactive: appactive activa otra aplicación, no a otra ventana dentro de Visual Basic, deberá asignarse el nombre de la aplicación objeto, esta será la aplicación que desea activarse, el nombre asignado será escrito de igual forma que aparece en el título (title bar).

Su aplicación es semejante a la de un manejador de tareas (task manager), o un manejador de programas (program manager), usualmente este comando precede al comando sendkey, para lograr activar la aplicación y poder así personalizar de una vez la aplicación.

Print: el comando print muestra texto sobre la pantalla o manda información a un impresor, cuando se utiliza para presentar información en la pantalla, este comando coloca texto o variables numéricas sobre un form o sobre una caja de figura, el texto aparecerá en la localización descrita por medio de coordenadas X, Y; Visual Basic maneja esto automáticamente, incrementando en la coordenada X un carácter; cuando una nueva línea es inicializada, la coordenada X toma un valor de cero y agrega una unidad a la coordenada Y, sin embargo estas coordenadas pueden ser accedidas directamente por el programador o el usuario.

Eof (función): esta función informa cuando se ha alcanzado el final de un archivo de esta manera el programador evita el tener que estar leyendo el archivo cuando ya no hay información. La aplicación de este comando es útil en archivos de bases de datos en los que hay que buscar información desde el principio del archivo hasta el final.

For...next: es uno de los comando más útiles en cualquier lenguaje de programación, debido a que crea un lazo que repetidamente ejecuta la o las instrucciones que se encuentran entre el comando For y el Next que lo acompaña. El numero de repeticiones del Loop es determinado por los números que se escriben a continuación del comando For.

7.10 FUNCIONAMIENTO DEL PROGRAMA DE CONTROL

En este capitulo se explicará el funcionamiento del software de control o salida comenzando primero con los flujogramas y luego con la explicación del programa desarrollado en visual basic para poder obtener el control de las variables para luego terminar con la codificación del programa.

7.10.1 Diagramas de flujo

A continuación se presentan los diagramas de flujo de las subrutinas que componen el programa comenzando con el diagrama de flujo del procedimiento con el cual se fijan las variables que determinan el numero de adquisiciones, el periodo de muestreo y los ajustes que el usuario desea tener con cada adquisición, seguidamente se presenta la subrutina que compara la adquisición de los datos con las variables fijadas y efectúa el control.

7.11 EXPLICACION DEL PROGRAMA

La lógica que controla el hardware del sistema se lleva a cabo mediante Visual Basic, un lenguaje de programación que tiene entre otras muchas ventajas el manejo de interfaces visuales con gran facilidad y es compatible con cualquier software que trabaja bajo Microsoft Windows; además posee la versatilidad de poder manejar los puertos seriales y paralelo de una PC o alguna otra tarjeta que se instale en el computador, tal como una tarjeta de modem o para el caso una interfase para el manejo de adquisición de datos.

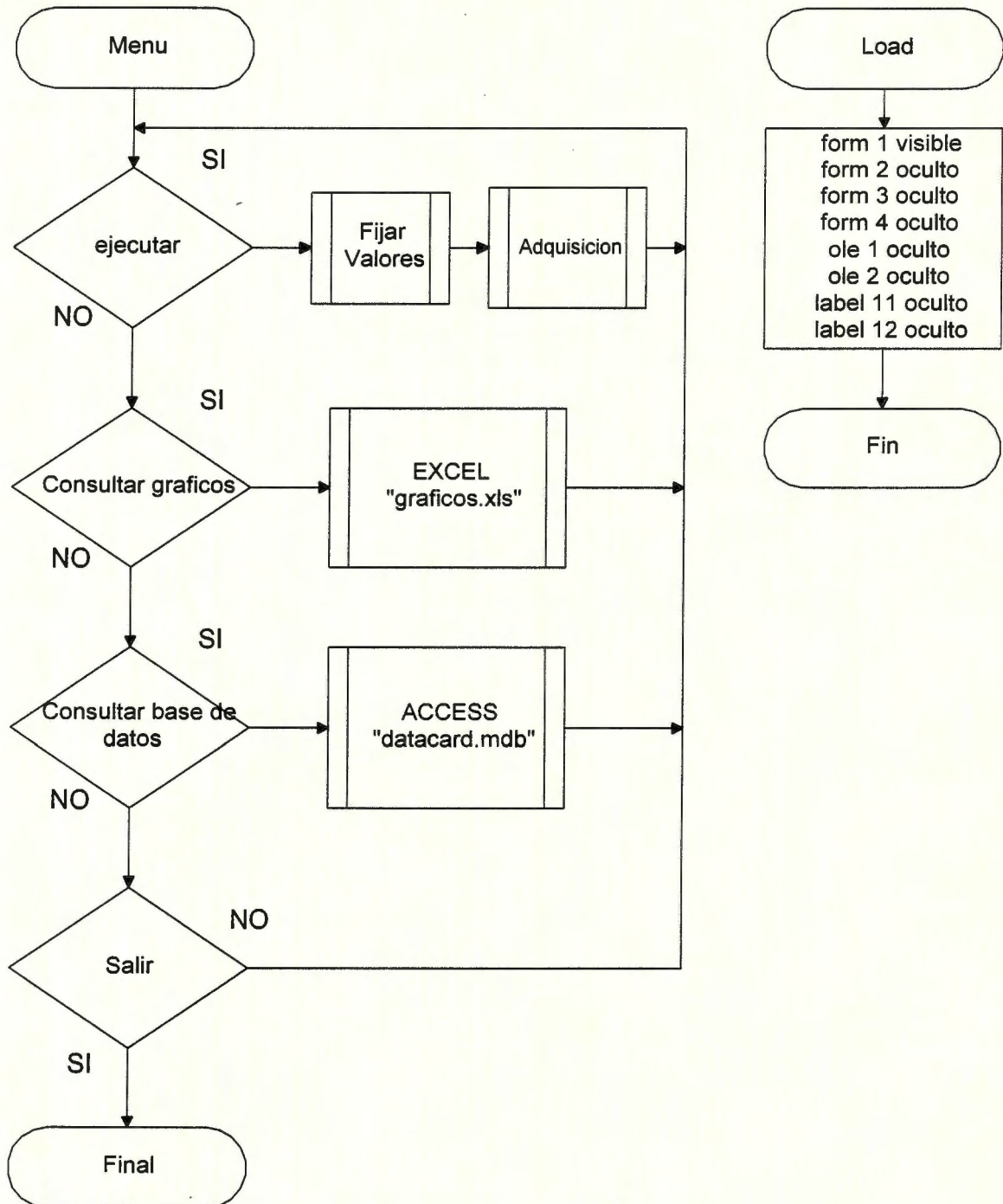
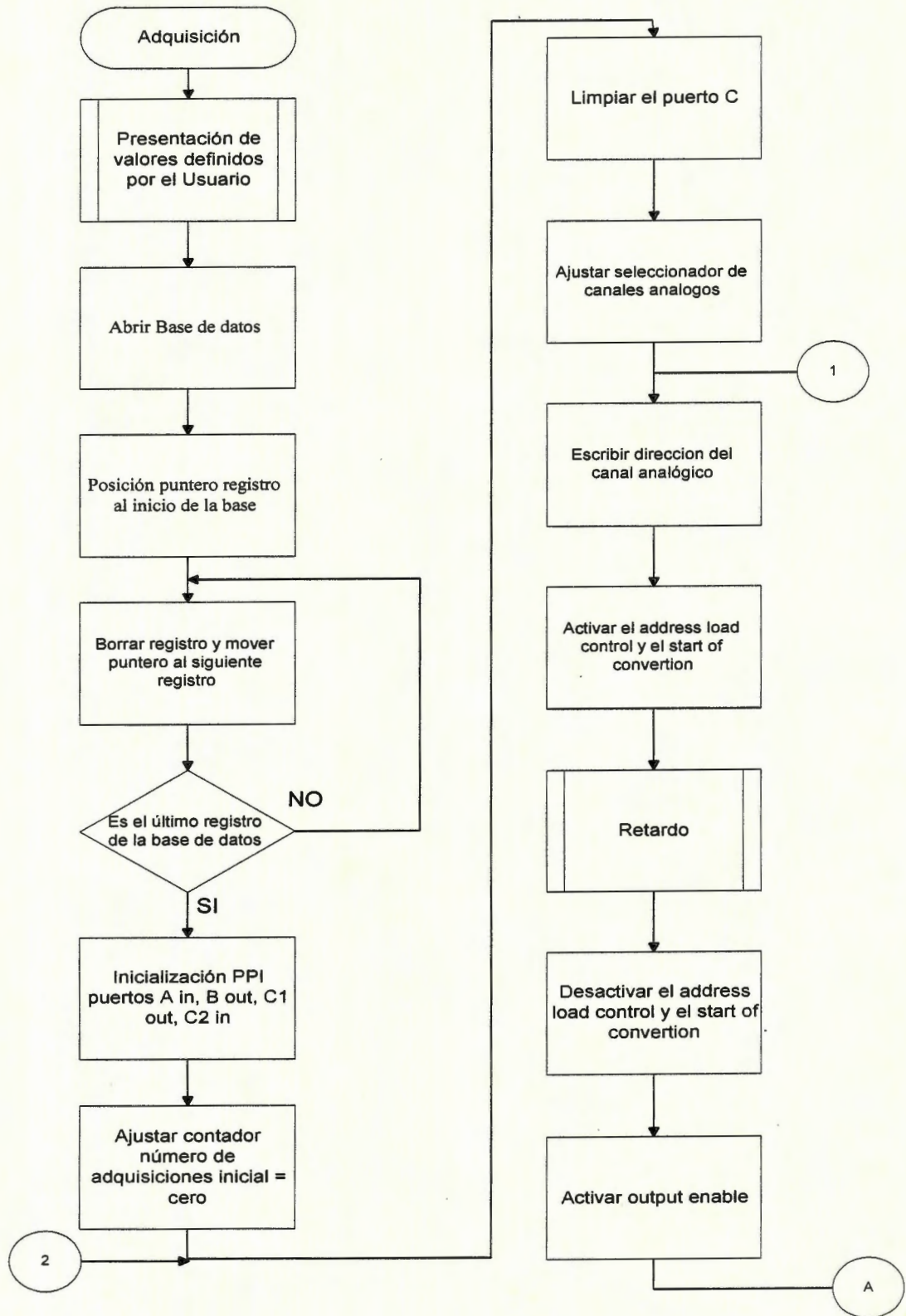
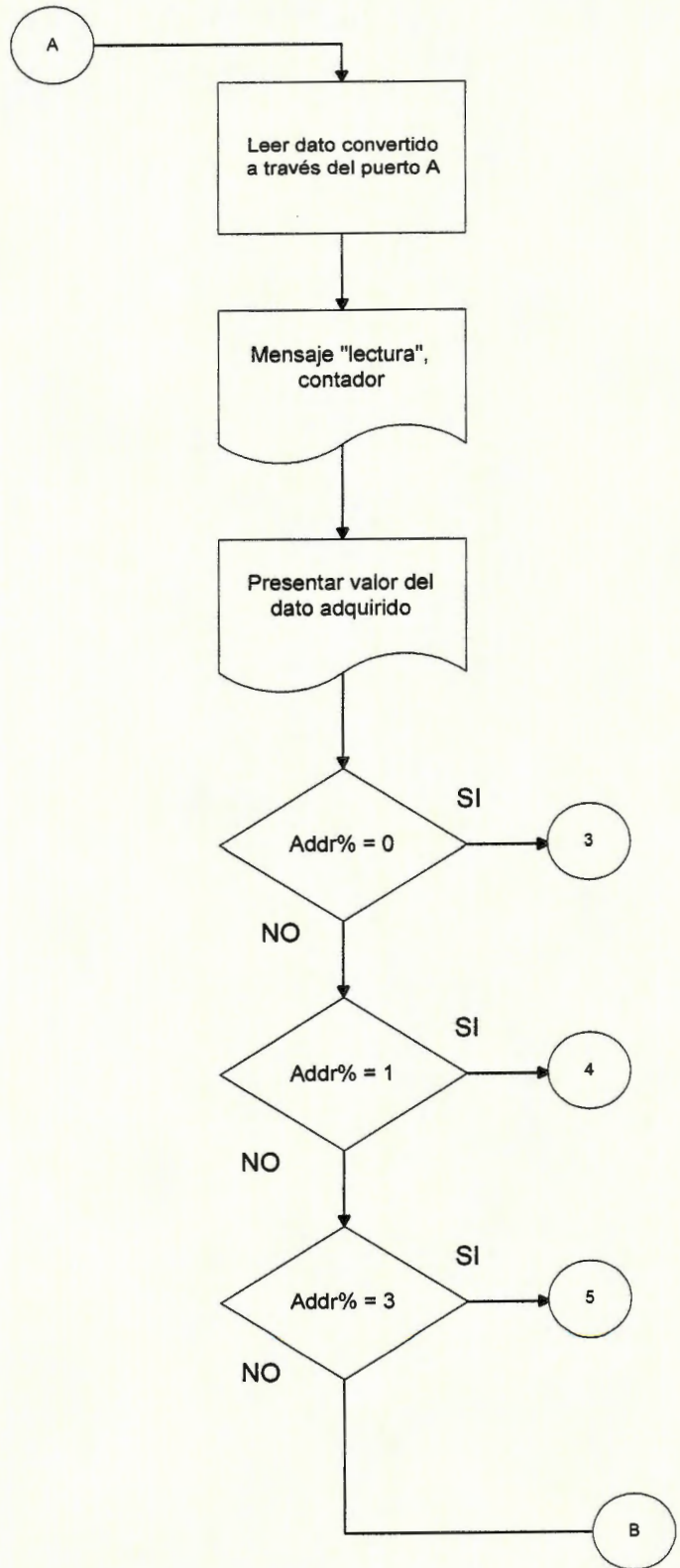
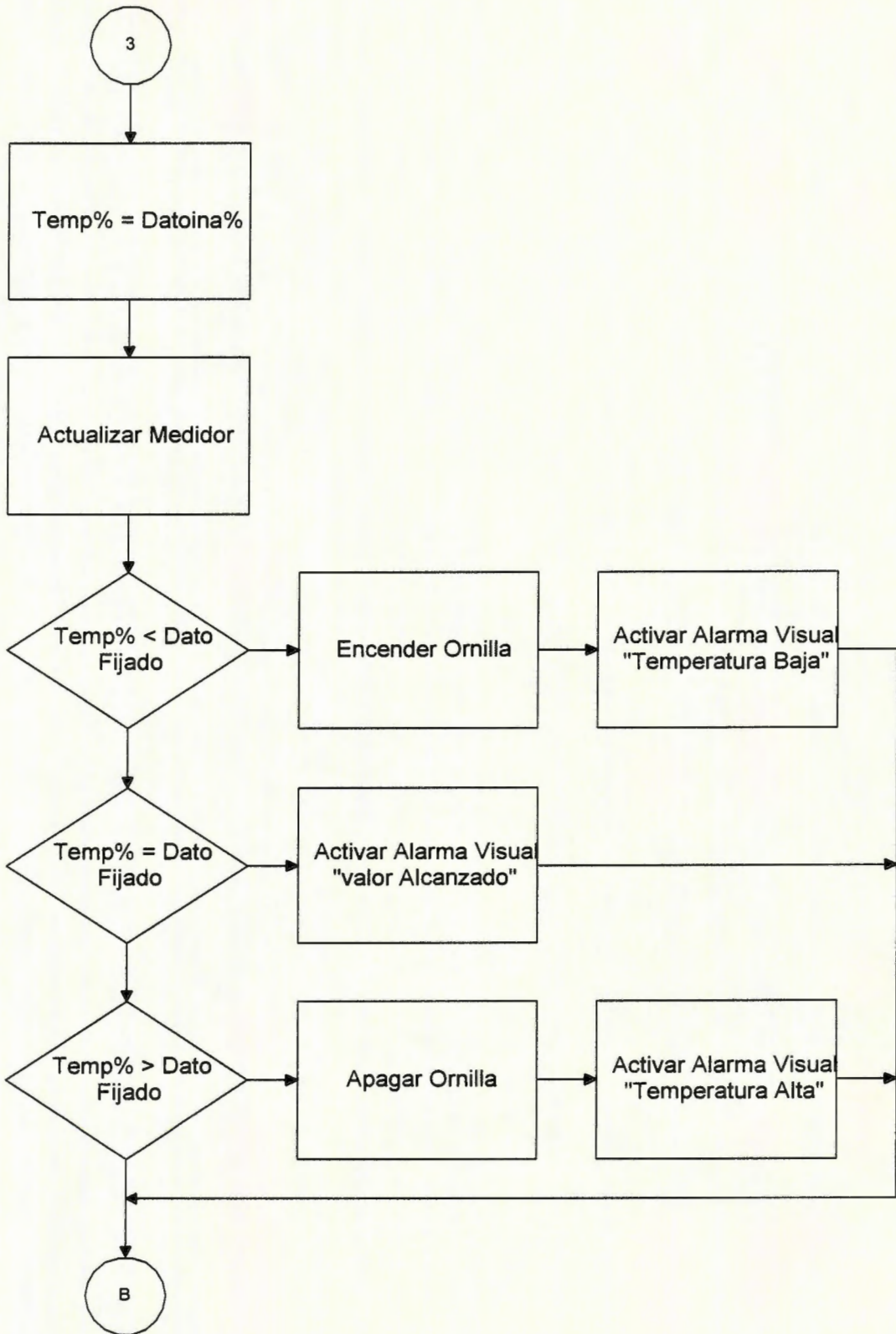
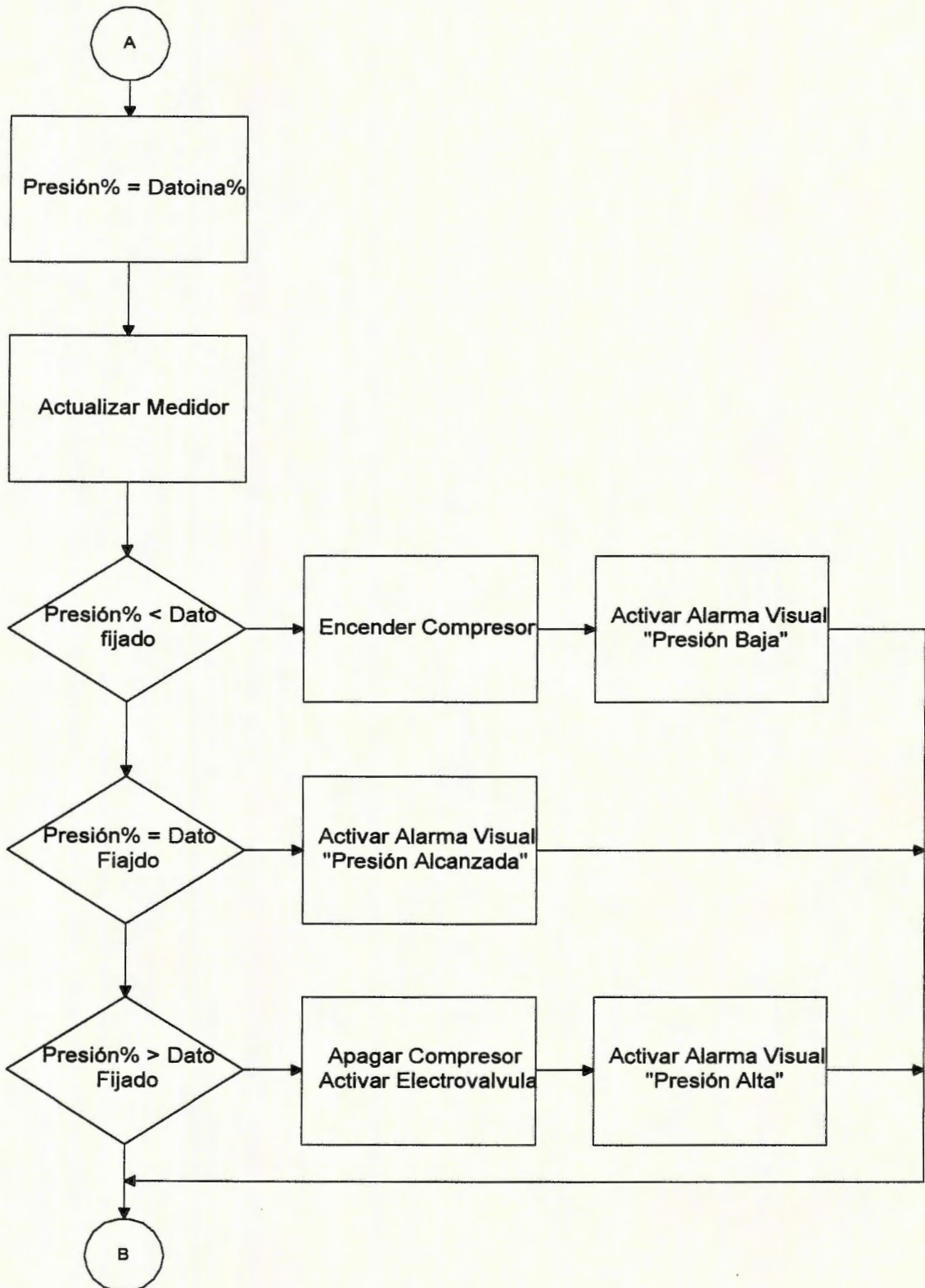


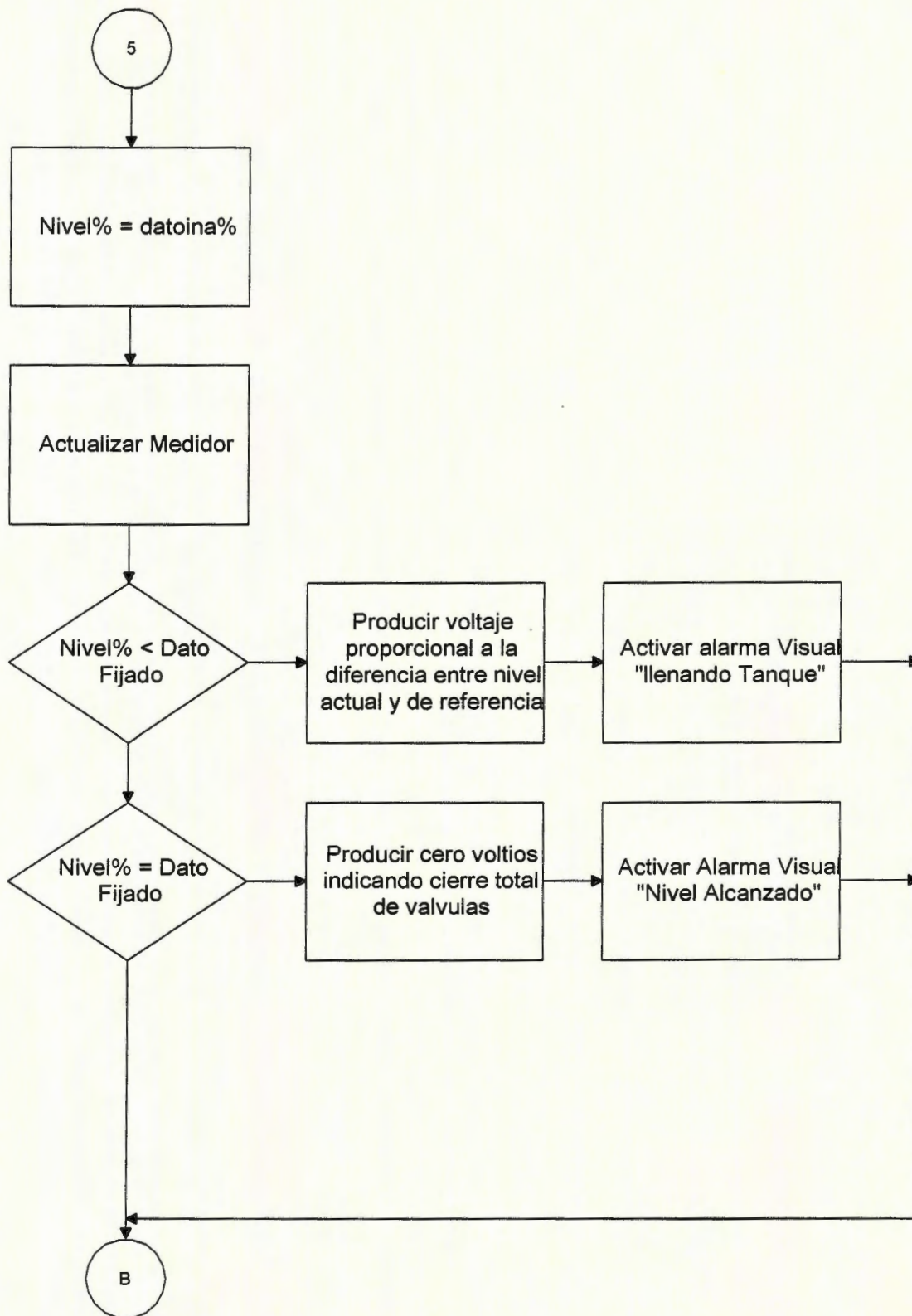
figura 20. Subrutina presenta y Load











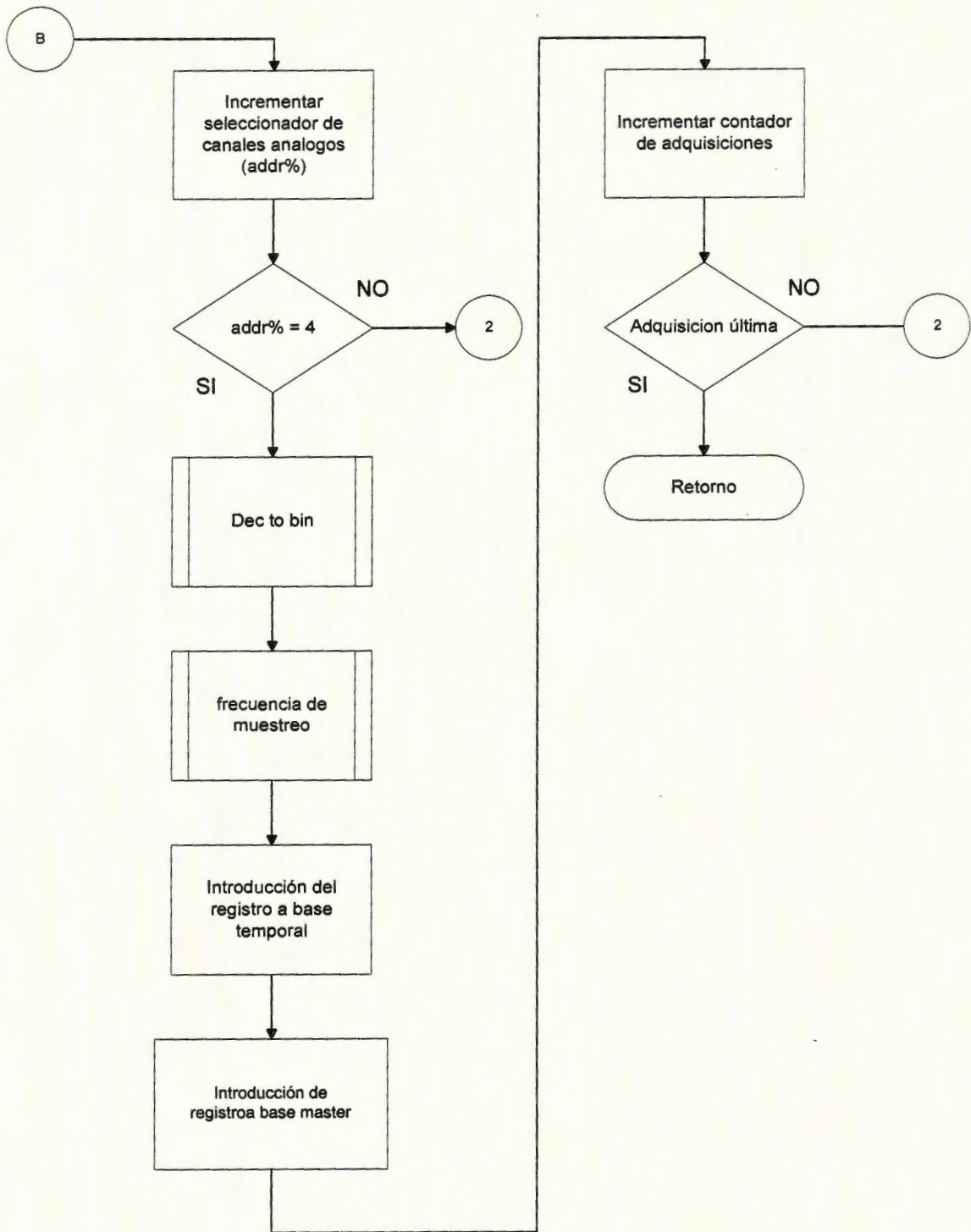


figura 21. Subrutina de adquisición

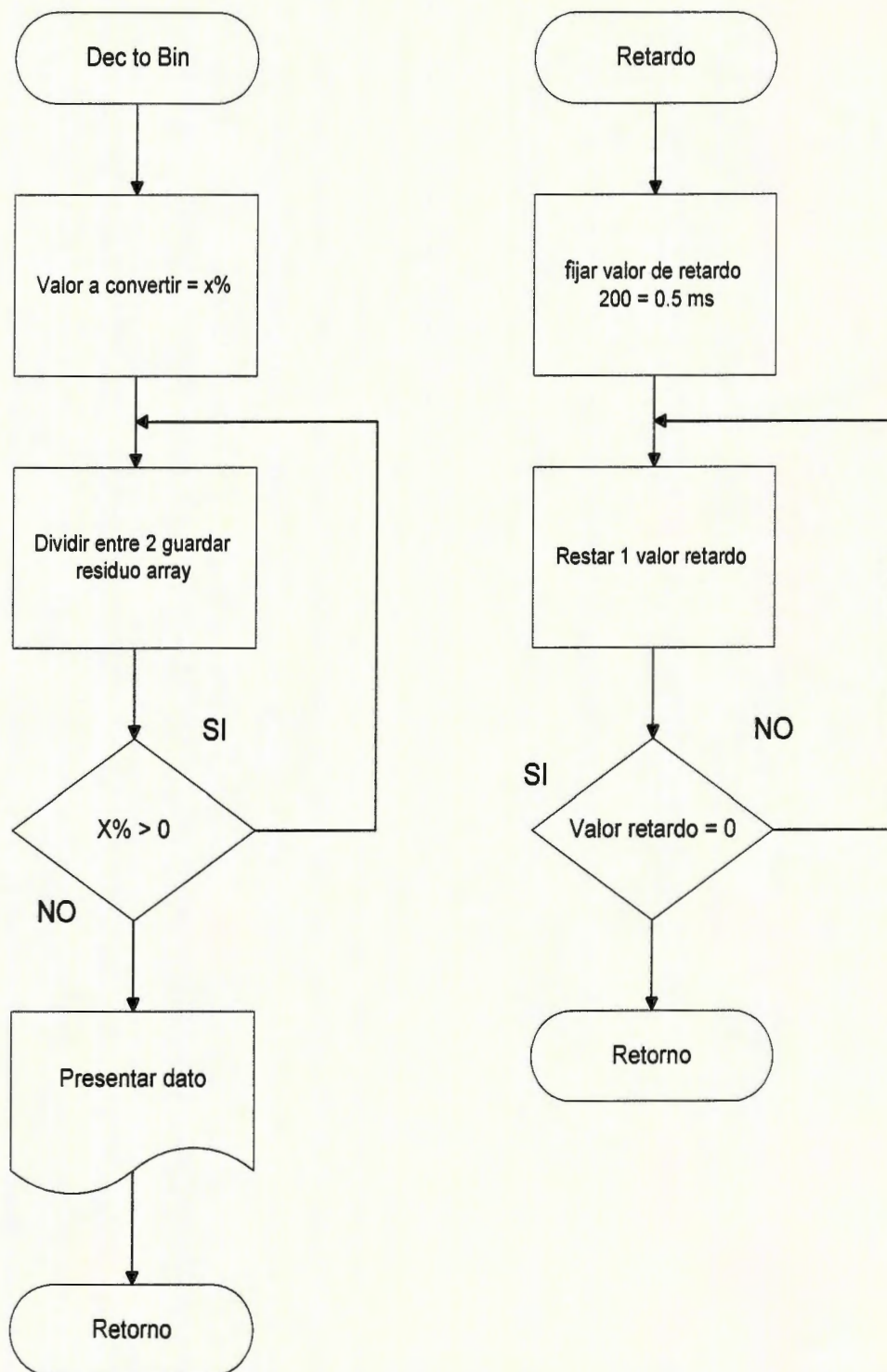


figura 22. Subrutina DEC to BIN y de retardo

La diferencia entre Visual Basic y la mayoría de programas es la facilidad con que se puede diseñar la pantalla, en lugar de ocupar un programa para hacer las pantallas de presentación (interfase). En Visual Basic, se dibujan las pantallas. Además cuando la interfase se ha dibujado los objetos (botones de comando , las cajas de texto y otros controles) que se han colocado en una ventana en blanco automáticamente reconocerán las acciones del usuario (eventos) tales como los movimientos del mouse, digitación de teclas y presión de los botones del mouse.

Después que se ha diseñado la interfase Visual Basic hace cualquier cosa como ocurre en un programa tradicional. Siempre se necesitará escribir códigos para que los objetos respondan a los eventos. Esto hace que Visual Basic sea totalmente diferente que los programas convencionales.

Los programas en lenguajes de programación convencionales corren en una forma top-down (de arriba hacia abajo) en el cual el programa inicia desde la primera línea y se mueve según el flujo del programa a las diferentes partes. La plataforma de Visual Basic es un arreglo de piezas independientes de código que son activadas por, y de forma tal responden solamente a los eventos programados

Cuando el programa inicia, se ejecutan los comandos que se encuentran en el objeto load en donde se establecen las condiciones iniciales tales como que objetos quedan visibles y que objetos quedan ocultos. El objeto inicialmente visible es el form nombrado "presenta" que contiene un menú de tipo pull-down (para jalar), en donde el usuario selecciona el procedimiento a ejecutar. Además se encuentran dos objetos ole y dos etiquetas inicialmente ocultos (ver figura 23).

Para iniciar la adquisición, bajo el submenu archivo se encuentra la opción "ejecutar", para finalizar el programa se encuentra la opción "salir". Si se necesita hacer una consulta del comportamiento del sistema, bajo el submenu consulta se encuentran las opciones de historial, base de datos y gráficos.

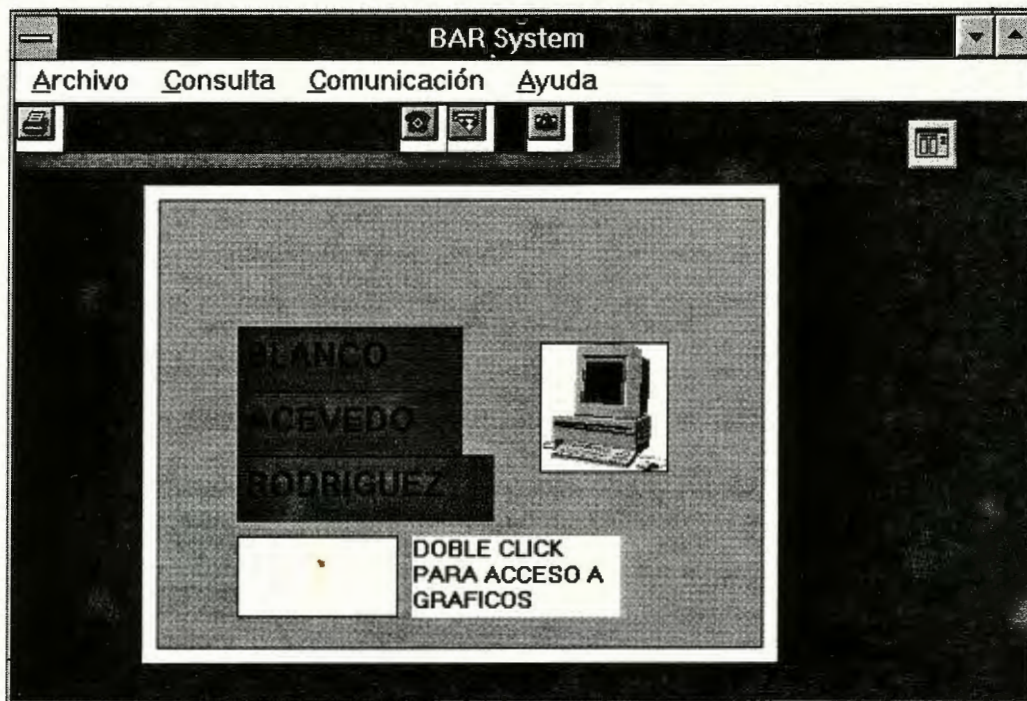


Figura 23. Pantalla de presentación

7.11.1 Opción ejecutar

La opción de ejecutar consta de dos forms al iniciar esta opción aparece el form settings, el cual consiste en una serie de cajas de texto y botones de comando para que el usuario pueda seleccionar e introducir los diferentes parámetros y valores necesarios para el sistema. Los valores a fijar son: número de adquisiciones, periodo de muestreo, y set points de las diferentes variables.

Cuando el form settings ha sido correctamente completado, el form lectura recibe el focus (control) y por medio de una acción click se inicia la adquisición de datos. En el form de la adquisición se tienen dos controladores de base de datos, tres medidores (gauges), cajas de texto donde se muestran los valores que el usuario fijó en el form anterior, un botón para salida del form actual y tres etiquetas (ver figura 24).

The screenshot shows a 'Settings' window with the following fields and options:

- Numero de adquisiciones:** A text input field with a minimum value of 1 and a maximum value of 200.
- Periodo de muestreo:** A radio button selection with options: 0.1 Seg., 0.5 Seg., 1.0 Seg., 1.5 Seg., and 2.0 Seg.
- Nivel:** A text input field with a minimum value of 4 (Lts) and a maximum value of 32 (Lts).
- Temperatura:** A text input field with a minimum value of 65°C and a maximum value of 350°C.
- Presión:** A text input field with a minimum value of 0 (PSI) and a maximum value of 16 (PSI).
- Control Continuo:** A checkbox.
- Iniciar Adquisicion:** A button to start the acquisition process.

figura 24. Form de fijarval

Inicialmente se abre la base de datos datacard.mdb, la cual esta programada para recibir el numero de registros que el usuario fija (máximo 200 adquisiciones por ejecución) que contendrá los datos adquiridos. La base de datos esta organizada en los siguientes campos: fecha, hora, temperatura, presión y nivel. A continuación se procede a eliminar los registros de la ejecución anterior y se almacenan en la base de datos datacard1.mdb donde se lleva el historial de todas las adquisiciones realizadas y tener así una base de datos limpia.

Los datos de las direcciones de los puertos son almacenados en las variables, puertoX% en donde X corresponde al puerto seleccionado en el PPI.

Se programa un contador para adquirir una cantidad determinada de datos (n-veces), dentro de el se encierra el programa que ejecuta la adquisición de los datos, y la rutina se repite n-veces. La rutina inicia escribiendo la palabra de control a el PPI a través del registro D, en donde el dato escrito fija el puerto A como entrada, el puerto B como salida, y el puerto C trabaja dividido de la siguiente manera: los cuatro bits menos significativos como salida y los cuatro mas significativos como entrada. En el puerto C se maneja la comunicación que gobierna la escritura y lectura que la computadora hace en el PPI , por lo

tanto para asegurar una condición inicial se escribe un 0 en el puerto C. La palabra de control a escribir es 152, la cual fija a el PPI en modo 0. A continuación se generan las direcciones del ADC0809 iniciando con la dirección 0 hasta la dirección 3, para leer los canales de entrada análogos. En esta etapa se habilitan los pines de Address Load Control (ALC) y Start Of Conversión (SOC), luego se produce una demora la cual mantiene las señales en alto. El estado de ALC y SOC es llevado a su condición original, asegurando así una transición de alto a bajo para iniciar la conversión (ver Anexo del diagrama de tiempo de ADC0809). Mientras la conversión toma lugar, se monitorea el pin de End Of Conversión (EOC) para determinar que el dato ya ha sido convertido y esta listo para ser leído del registro del ADC. Para que el dato convertido pueda ser leído por el PPI a través del puerto A, es necesario habilitar el pin Output Enable (Outen) del ADC. El dato adquirido es almacenado en una variable nombrada datoina%. Dependiendo de la dirección del canal análogo, se asigna una variable al datoina%, de la siguiente forma: canal cero temp%=datoina%, canal 1 presion%=datoina% y canal 3 nivel%=datoina%. Cada una de estas variables actualiza el gauge y el campo respectivo en la base de datos. Después que el dato ha sido leído y las bases han sido actualizadas se deshabilita el Outen y se procede a llamar a la subrutina bin la cual muestra en formato binario los datos adquiridos en cada entrada, esto se debe a que los datos son leídos por el programa en formato decimal. Dependiendo de la variable que esta siendo adquirida, esta es comparada con el set point y dependiendo de si la variable se encuentra abajo del set point se procede a tomar una acción o control de la siguiente forma:

Para el caso de addr%=0, la variable adquirida es temperatura, si el valor es menor que el set point, se ejecuta el comando de salida de datos para gobernar un relé que energiza una resistencia y de esta forma producir calor para elevar la temperatura hasta el set point. Cuando se alcanza el valor de temperatura programado, una instrucción de salida desconecta el relé y la resistencia deja de generar calor, para cada caso de comparación, se

presenta una alarma visual para que el usuario tenga una retroalimentación del comportamiento del sistema.

Para el caso de $addr\%=1$, la variable adquirida es presión, si el valor es menor que el set point, se ejecuta el comando de salida de datos para gobernar un relé que energiza un compresor para elevar la presión dentro de una cámara “hermética” cuando se alcanza el valor de presión programado, una instrucción de salida desconecta el relé, para apagar el compresor. Si se fija un valor de presión menor a la presión existente dentro de la cámara, se energiza un relé para poder alimentar una electroválvula que deja escapar la presión hasta que se alcanza el valor fijado. Para cada caso de comparación, se presenta una alarma visual (en el software) para que el usuario tenga una retroalimentación del comportamiento del sistema.

Para el caso de $addr\%=2$, la variable adquirida es nivel, si el valor es menor que el set point, se produce un voltaje proporcional a medida que el tanque se esta llenando , para el caso que el tanque esta vacío se produce un voltaje máximo, a medida el tanque se va llenando el voltaje va reduciendose hasta que la electroválvula se cierra, esto se lleva acabo a través de una ecuación implementada por medio del software, la cual consiste de una resta de el valor de set point menos el valor leído en el momento de la adquisición, resultando una salida con un valor de voltaje que representa el comportamiento proporcional para cada posible resultado de la resta efectuada.

El siguiente paso consiste en fijar un periodo de muestreo que tiene un valor entre un rango de 0.1 hasta 2 Segundos, que puede ser fijado por el usuario. Seguidamente se introducen los campos a la base de datos temporal (datacard.mdb) y a la base de datos que contiene el historial de todas las adquisiciones (datacar1.mdb). El proceso se repite n veces (n =dato fijado por el usuario). Si se desean adquisiciones extras el usuario puede hacer un click en el form tantas veces como sea necesario. Para regresar al form donde se encuentra el menú principal existe un botón de comando etiquetado salir (ver figura 25).

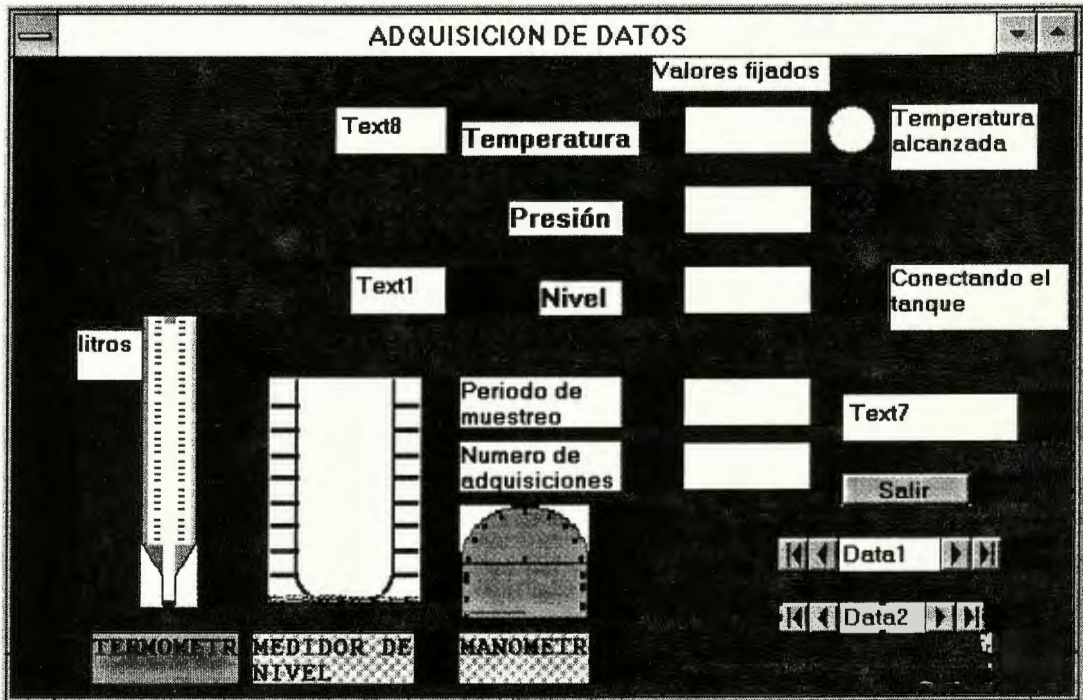


Figura 25. Pantalla de adquisición de datos

7.11.2 Opción salir

Es un comando end simple, el cual finaliza la sesión cerrando todos los forms y regresando al sistema operativo Windows.

7.11.3 Opción consulta

Con esta opción se puede elegir la representación de los datos por medio de un gráfico de tiempo versus valor o por medio de una base de datos donde se han registrado los datos adquiridos. La opción de gráficos hace visible un objeto de control OLE el cual al recibir un evento de doble click abre el archivo en Excel "graficos.xls", en donde se encuentran los datos adquiridos en forma de hoja electrónica junto con un macro que lleva a cabo la función de seleccionar los datos y dar las instrucciones de graficar y mostrar.

Dentro de Excel, el usuario puede llevar a cabo funciones propias de la aplicación tales como: imprimir el gráfico, ampliar un área de interés, guardar la información en un archivo, etc.

La opción de base de datos al igual que la de gráficos, hace visible un control OLE, el cual al recibir un evento doble click abre el archivo en Acces "datacard.mdb" en donde se encuentra un macro que lleva a cabo la función de presentar la base de datos y transferir la información en un formato de hoja electrónica a la aplicación de Excel. Dentro de Acces el usuario puede llevar a cabo funciones propias de la aplicación tales como: hacer un ordenamiento (sort), elaborar un reporte, imprimir un rango de registros, sacar un promedio, etc.

La opción historial presenta una base de datos que contiene todas las adquisiciones efectuadas desde el inicio de una sesión (usualmente el inicio del programa). Al igual que las dos opciones anteriores, un control OLE y una etiqueta aparecen cuando se selecciona esta opción.

7.11.4 Subrutina bin

La función de esta subrutina es convertir los datos adquiridos a un formato binario por medio de un proceso de división sucesiva entre dos del dato adquirido, donde el residuo es almacenado en un array que tiene capacidad para ocho dígitos. A continuación se procede a ensamblar el dato binario leyendo el array en forma descendente y leyendo el resultado uno a continuación del otro.

7.12 PROGRAMA PRINCIPAL

A continuación se detalla la codificación completa del programa dividida en subrutinas para cada objeto y su respectivo evento.

7.12.1 Presenta

* Se incluyen en esta subrutina los eventos de menu, load y click.

Sub acerca ()

form4.Visible = True ' En este form se encuentra información del programa

form4.SetFocus

End Sub

Sub Form_Load () ' Establece las condiciones iniciales de los objetos y forms

form4.Visible = False

form1.Visible = True

form2.Visible = False

form3.Visible = False

ole1.Visible = False

label1.Visible = False

label2.Visible = False

ole2.Visible = False

label3.Visible = False

ole3.Visible = False

End Sub

Sub mnuarchivoitem_Click (Index As Integer)

Select Case Index

Case 0 ' Opción de ejecutar

form1.Visible = False

form3.Visible = True

form3.SetFocus

Case 1 ' Opción salir

```
End  
End Select  
End Sub
```

Sub mnuAyudaitem_Click (Index As Integer)

```
acerca 'Opción de consulta de versión, programadores del software  
End Sub
```

Sub mnuconsultaitem_Click (Index As Integer)

```
Select Case Index  
Case 0 'Opción de consulta de graficos  
If ole1.Visible = True Then  
label1.Visible = False  
ole1.Visible = False  
label2.Visible = True  
ole2.Visible = True  
Show  
ole2.Class = "excel.chart.5"  
ole2.SourceDoc = "c:\vb\graficos.XLS"  
ole2.Action = 1  
Else MsgBox "Consultar primero Base de Datos"  
End If  
Case 2 ' Opción de consulta de base de datos temporal  
label2.Visible = False  
ole2.Visible = False  
label1.Visible = True  
ole1.Visible = True
```

Show

```
ole1.Class = "access.database.2"
```

```
ole1.SourceDoc = "c:\vb\datacard.mdb"
```

```
ole1.Action = 1
```

```
'AppActivate "microsoft access"
```

Case 3 ' Opción de consulta de base de datos permanente (historial)

```
label3.Visible = True
```

```
ole3.Visible = True
```

Show

```
ole3.Class = "access.database.2"
```

```
ole3.SourceDoc = "c:\vb\dataacar1.mdb"
```

```
ole3.Action = 1
```

```
End Select
```

```
End Sub
```

Sub Picture1_Click ()

```
CMDIALOG1.Action = 5 ' Opción de impresión
```

```
End Sub
```

Sub Picture8_Click ()

```
acerca ' Opción de consulta de versión, programadores del software
```

```
End Sub
```

7.12.2 Inivar

*Inicializa las variables con condición de global (para poder ser utilizadas por todos los *forms)

```
Global frec
```

Global numadquisic As Integer

Global nivelmin As Integer

Global nivelmax As Integer

Global temperval As Integer

Global presval As Integer

Global continuo As Integer

7.12.3 Fijarval

* Presenta el form para poder introducir los datos de set point y periodo de adquisición.

Sub Command3D1_Click () ' Boton de iniciar adquisición

If text2.Text = "" Or text3.Text = "" Or text4.Text = "" Or text5.Text = "" Or frec = 0 Then

MsgBox "Asegurese de no dejar casillas sin valor"

Else

If check1.Value = 1 Then

continuo = 1000

Else

continuo = 1

End If

form3.Visible = False

form2.Visible = True

form2.SetFocus

End If

MsgBox "Haga Click en el area verde para iniciar adquisición"

End Sub

Sub Form_Load ()

```
frec = 0
text4.Visible = False
label7.Visible = False
End Sub
```

* Las opciones a continuación, se refieren a los diferentes botones de selección
* en el form y determinan la secuencia de introducción de datos

```
Sub Option1_Click () 'Habilita la introducción del dato de nivel
MsgBox "Valor minimo 0, valor máximo 32 en pasos de 4", 64, "NIVEL"
label3.Visible = True
label4.Visible = True
text2.Visible = True
text2.SetFocus
End Sub
```

```
Sub Option2_Click () 'Habilita la introducción del dato de temperatura
MsgBox "Introducir valores enteros", 64, "TEMPERATURA"
text3.Visible = True
text3.SetFocus
label6.Visible = True
End Sub
```

```
Sub Option3_Click () 'Habilita la introducción del dato de presión
text4.Visible = True
text4.SetFocus
label7.Visible = True
End Sub
```

```
Sub Option4_Click () 'Habilita la introducción del dato de periodo de muestreo  
frame1.Visible = True  
End Sub
```

```
Sub Option9_Click ()  
frame1.Visible = True  
End Sub
```

```
Sub Text2_KeyPress (keyascii As Integer)  
* Caja de texto para introducir el dato de nivel  
If keyascii = 13 Then ' Si se presionó la tecla de <ENTER>  
nivelmax = Val(LTrim(text2.Text) & "&")  
If nivelmax > 32 Then  
MsgBox "El valor no debe exceder de 32", 16, "Nivelmax"  
Else  
option2.SetFocus  
End If  
End If  
End Sub
```

```
Sub Text3_KeyPress (keyascii As Integer)  
* Caja de texto para introducir el dato de temperatura  
If keyascii = 13 Then ' Si se presionó la tecla de <ENTER>  
temperval = Val(LTrim(text3.Text) & "&")  
If temperval < 65 Or temperval > 350 Then  
MsgBox "El valor debe estar en un rango de 65 a 350", 16, "Temp Value"
```

```

Else
option3.SetFocus
End If
End If
End Sub

```

Sub Text4_KeyPress (keyascii As Integer)

```

* Caja de texto para introducir el dato de presión
If keyascii = 13 Then ' Si se presionó la tecla de <ENTER>
presval = Val(LTrim(text4.Text) & "&")
If presval < 0 Or presval > 16 Then
MsgBox "El valor debe estar en un rango de 0 a 16", 16, "Temp Value"
Else command3d1.SetFocus
End If
End If
End Sub

```

Sub text5_GotFocus ()

```

* Texto de información antes de iniciar la introducción de datos
MsgBox "Presionar <Enter> después de introducir cada dato", 48, "Importante"
End Sub

```

Sub Text5_KeyPress (keyascii As Integer)

```

* Caja de texto para introducir numero de adquisiciones por corrida
If keyascii = 13 Then ' Si se presionó la tecla de <ENTER>
numadquisic = Val(LTrim(text5.Text) & "&")
If numadquisic < 1 Or numadquisic > 200 Then

```

```
MsgBox "El valor debe estar en un rango de 1 a 200", 16, "Num Adquisic"
```

```
Else
```

```
option4.SetFocus
```

```
End If
```

```
End If
```

```
End Sub
```

Sub valor_Click (index As Integer)

* Se asigna el valor de periodo de muestreo para cada selección del boton respectivo

```
Select Case index
```

```
Case 0 ' Boton de 0.1 segundos
```

```
frec = .1
```

```
option1.SetFocus
```

```
Case 1 ' Boton de 0.5 segundos
```

```
frec = .5
```

```
option1.SetFocus
```

```
Case 2 ' Boton de 1.0 segundos
```

```
frec = 1
```

```
option1.SetFocus
```

```
Case 3 ' Boton de 1.5 segundos
```

```
frec = 1.5
```

```
option1.SetFocus
```

```
Case 4 ' Boton de 2.0 segundos
```

```
frec = 2
```

```
option1.SetFocus
```

```
End Select
```

```
End Sub
```

7.12.4 Lectura

* Subrutina de adquisición. Inicia fijando el puerto paralelo para comunicación

* bidireccional

```
Declare Sub agoutp Lib "apiguide.dll" (ByVal portid%, ByVal outval%)
```

```
Declare Function aginp% Lib "apiguide.dll" (ByVal puerto%)
```

Sub bin (x%)

* Subrutina de conversión de decimal a binario donde X%= dato adquirido

```
Print "DECIMAL", x%
```

```
I = 0
```

```
ReDim RES(255)
```

```
Do While x% > 0
```

```
    I = I + 1
```

```
    RES(I) = x% Mod 2
```

```
    x% = Int(x% / 2)
```

```
Loop
```

```
Print "BINARIO",
```

```
For a = I To 1 Step -1
```

```
    Print RES(a);
```

```
Next a
```

```
Print Chr$(13)
```

```
End Sub
```

Sub Command1_Click ()

* Boton de comando para salir del form de lectura. Transfiere el focus a presenta

```
form2.visible = False
```

```
form1.visible = True
form1.SetFocus
End Sub
```

Sub Form_Click ()

* Form de adquisición y comparación que se activa al hacer un click

```
For repetir = 1 To continuo ' Adquisición en modo continuo
text2.Text = presval      'caja de texto con valores del usuario
text3.Text = nivelmax    'caja de texto con valores del usuario
text4.Text = temperval   'caja de texto con valores del usuario
text5.Text = frec        'caja de texto con valores del usuario
text6.Text = numadquisic 'caja de texto con valores del usuario
data1.Recordset.MoveFirst
    Do While Not data1.Recordset.EOF
        data1.Recordset.Delete
        data1.Recordset.MoveNext
    Loop
puertoa% = 608
puertob% = 609
puertoc% = 610
puertod% = 611
```

```
For CONTADOR = 1 To numadquisic 'Dato de numero de adquisiciones
data1.Recordset.AddNew
data2.Recordset.AddNew
Cls
Print "LECTURA NUMERO ", CONTADOR
```

```

For addr% = 0 To 2 Step 2
  agoutp puertoc%, addr%
  sumaddrs% = 4 + addr%   'activar ALC y SOC
  agoutp puertoc%, sumaddrs%
  For x = 1 To 200      'demora
  Next x
  agoutp puertoc%, addr%   'desactivar ALC y SOC
  outen% = 8
  agoutp puertoc%, outen% 'activar OUTEN
  For x = 1 To 200
  Next x      'demora
  datoina% = aginp(puertoa%) 'lectura puerto A
  outen% = 0
  agoutp puertoc%, outen% 'desactivar OUTEN
  Print datoina%
  If addr% = 0 Then      'temperatura
  temp1% = datoina%
  temp% = (2 * temp1%) + 65   'formula para pasar de dato a temperatura
  text8.Text = temp%      'caja de texto con dato de temperatura
  If temp% < temperval Then  'Si la temperatura < valor fijado
  Print "ENCENDER ORNILLA"
  hornon% = 1            'Encender horno
  shape1.visible = False
  label15.visible = True
  shape7.visible = True
  label16.visible = False
  shape6.visible = False

```

```

label14.visible = False
End If

If temp% > temperval Then      'Si la temperatura > valor fijado
Print "APAGAR ORNILLA"
hornon% = 0                    'Apagar horno
shape1.visible = False
label14.visible = True
shape6.visible = True
label16.visible = False
shape7.visible = False
label15.visible = False
End If

If temp% = temperval Then      'Si la temperatura = valor fijado
shape1.visible = True          'Activa alarma
label16.visible = True
shape6.visible = False
label14.visible = False
shape7.visible = False
label15.visible = False
End If

bin datoina%
gauge1.Value = temp%
End If

If addr% = 2 Then              'Variable adquirida = presión
pres% = datoina%
If pres% < presval Then        'Si la presión < valor fijado

```

```

Print "ENCENDER compresor"
compon% = 2      'Encender compresor
compoff% = 0
shape2.visible = False
End If
If pres% > presval Or pres% = presval Then 'Si la presión < valor fijado

Print "APAGAR compresor"
compon% = 0
compoff% = 4
shape2.visible = True
Beep
End If
gauge2.Value = datoina%
End If
bin datoina%    'conversion DEC to BIN
Next addr%
addr% = 3
agoutp puertoc%, addr%
datoina% = aginp(puertoa%) 'lectura puerto A
For y = 1 To 200
Next y
nivel% = datoina%
bin datoina%
Select Case nivel%
Case 0
nive = 0

```

gauge3.Value = 0

Case 1

Print 4

nive = 4

gauge3.Value = 4

Case 3

Print 8

nive = 8

gauge3.Value = 8

Case 7

Print 12

nive = 12

gauge3.Value = 12

Case 15

Print 16

nive = 16

gauge3.Value = 16

Case 31

Print 20

nive = 20

gauge3.Value = 20

Case 63

Print 24

nive = 24

gauge3.Value = 24

Case 127

Print 28

```

nive = 28
gauge3.Value = 28
Case 255
Print 32
nive = 32
gauge3.Value = 32
End Select

    proporcion = (nivelmax - nive) * 3.75
    salida% = proporcion And 248
out% = hornon% Or compon% Or compoff% Or tanon% Or tanoff%
text7.Text = out%      ' Salida de palabra de control de los actuadores
agoutp puertob%, out%
periodo = frec * 350000
For jocker = 1 To periodo      ' Periodo fijado por el usuario
Next jocker

* Los pasos siguientes, son para la actualización de la base de datos master & temporal
data1.Recordset.Fields("Fecha") = Date$
data1.Recordset.Fields("Hora") = Time$
data1.Recordset.Fields("Temperatura") = temp%
data1.Recordset.Fields("Presion") = pres%
data1.Recordset.Fields("Nivel") = nivel%
data1.Recordset.Update
data2.Recordset.Fields("Fecha") = Date$
data2.Recordset.Fields("Hora") = Time$
data2.Recordset.Fields("Temperatura") = temp%
data2.Recordset.Fields("Presion") = pres%
data2.Recordset.Fields("Nivel") = nivel%

```

```
data2.Recordset.Update
```

```
Next CONTADOR
```

```
Next repetir
```

```
End Sub
```

Sub Form_Load ()

* Fija las condiciones iniciales del form adquisición

```
puertoa% = 608
```

```
puertob% = 609
```

```
puertoc% = 610
```

```
puertod% = 611
```

```
datoc% = 0 'iniciar puerto C
```

```
datod% = 152 'ctrl word(Ain,Bout,Clowout,Cupin)
```

```
agoutp puertod%, datod%
```

```
agoutp puertoc%, datoc%
```

```
shape1.visible = False
```

```
shape2.visible = False
```

```
shape3.visible = False
```

```
shape4.visible = False
```

```
shape5.visible = False
```

```
shape6.visible = False
```

```
shape7.visible = False
```

```
label11.visible = False
```

```
label12.visible = False
```

```
label13.visible = False
```

```
label14.visible = False
```

```
label15.visible = False
```

```
label16.Visible = False
```

```
End Sub
```

7.12.5 About

*Form de información acerca del software

```
Sub Command1_Click ()
```

```
form1.Visible = True
```

```
form1.SetFocus
```

```
form4.Visible = False
```

```
End Sub
```

CONCLUSIONES

Del presente trabajo de graduación se concluye:

- La manera de garantizar la confiabilidad en un sistema de adquisición de datos para el control automático de variables, es asegurando que la frecuencia de muestreo del sistema sea al menos el doble de la frecuencia de cambio de la variable en observación.

- Para evaluar un sistema de adquisición de datos es necesario considerar principalmente los siguientes criterios: estabilidad, tiempo de respuesta, y precisión.

- El rango de las variables a sensar esta sujeto a los limites de medición de cada sensor y el control que se obtenga de estos esta sujeto a la capacidad de los actuadores que intervengan en el sistema.

- Es necesario sentar una base informativa propia acerca de esta clase de proyectos debido a las limitaciones existentes por falta de información en nuestro medio.

- El uso de computadores personales provee una mayor rapidez y exactitud para el procesamiento de datos y obtención de resultados, lo cual hace al sistema más confiable en procesos industriales.

- Para que el comportamiento del sistema sea fiel deberá garantizarse la linealidad de los sensores, actuadores y componentes discretos del hardware del sistema.

GLOSARIO TECNICO

ACTUADORES: Dispositivos a través de los cuales se acciona un elemento para corregir una variable física.

AT: Advanced Technology (tecnología avanzada).

BASE DE DATOS: Es una colección de archivos interrelacionados, arreglados en una forma altamente organizado, que es creado y manejado por un sistema manejador de base de datos.

BIT: Unidad de almacenamiento mas pequeña de una computadora.

BYTE: Palabra compuesta por ocho bits.

COMPATIBILIDAD: Disponibilidad para trabajar en conjunto.

CONVERTIDOR A/D: Dispositivo que transfiere información de formato análogo a digital.

CONVERTIDOR D/A: Dispositivo que transfiere información de formato digital a análogo.

HARDWARE: Equipo físico que constituye un sistema de computadora.

INTERFASE: Interconexión e interrelación entre dos elementos, dos aplicaciones o el usuario y una aplicación.

MICROPROCESADOR: Unidad central de proceso (CPU).

PC: Personal Computer (computadora personal).

PUERTO PARALELO: Canal de entrada y salida utilizado para conectar periféricos que utilizan un protocolo paralelo (uno o mas bytes simultáneamente).

REGISTRO: Área de almacenamiento de datos.

RUIDO: Señales de naturaleza extraña que interfieren con las transmisiones eléctricas.

SEÑAL ANALOGICA: Señales que pueden tener un numero infinito de valores.

SEÑAL DIGITAL: Señales que tienen un valor discreto y finito.

SISTEMA DE CONTROL AUTOMATICO DE LAZO CERRADO: Sistemas de control retroalimentado.

SOFTWARE: Secuencia lógica de comandos que forman un programa de computadora.

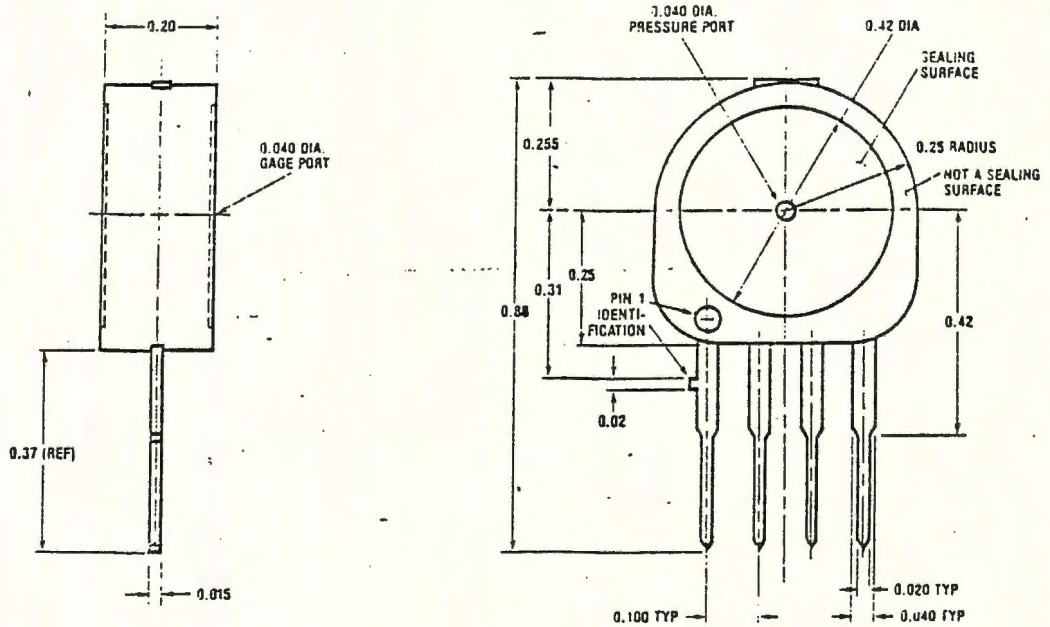
TRANSDUCTOR: Dispositivo el cual convierte un tipo de energía a otra diferente.

TTL: Tecnología de fabricación de circuitos integrados que utiliza transistores BJT como unidad básica.

VAC: Voltaje de naturaleza alterna.

TYPICAL PHYSICAL DIMENSIONS

All dimensions shown are in inches. Tolerances are ± 0.01 inches except where otherwise stated.



ORDERING INFORMATION

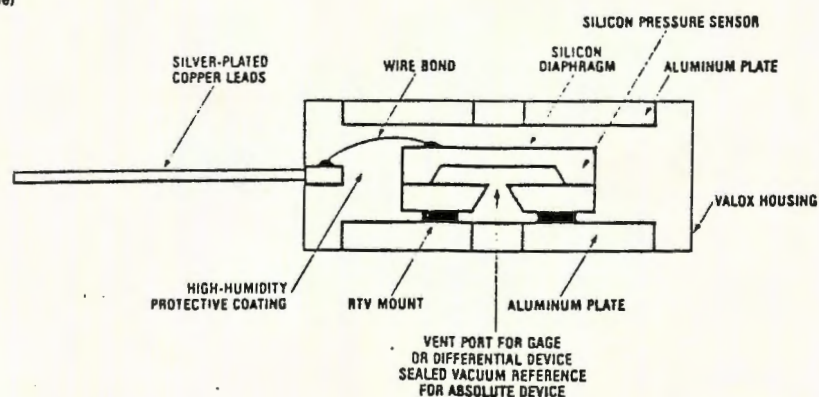
To order, use the following part numbers:
 Absolute Pressure SPX50A
 Differential (Gage) Pressure SPX50D
 Order through Sensym inc., or your local representative or distributor.

ADDITIONAL SPX SERIES INFORMATION

For information on the SPX100(A, D) and SPX200A, 0 to 15 psi and 0-30 psi parts respectively, please contact the Sensym factory.

PHYSICAL CONSTRUCTION

(Drawing not to scale)



Mechanical Notes:

Note 1: This package is intended to have pressure applied to the top side of the device. The fact is that the device could be inserted backward, with pressure going to the back side of the silicon diaphragm. In this configuration, the device will function identically to the forward configuration. The potential problem would be that with high-pressure or repeated cycling, the device might fail due to the soft RTV mount. If the device is operated with pressure applied to the back side of the diaphragm, the supply voltage polarity must be reversed to get the output polarity shown.

Note 2: The user should consider the compatibility of the fluid with the materials shown in the Physical Construction drawing.

Electrical Notes:

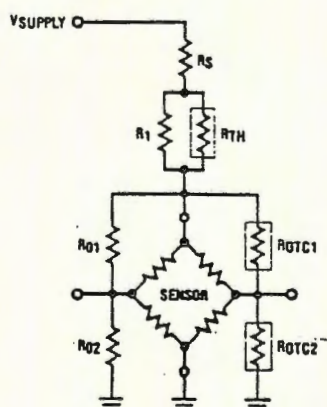
Note 1: The device is ratiometric to the supply voltage. Changes in supply voltage will cause changes in the transfer curve. Both offset voltage and full-scale span will change with changes in supply voltage.

Note 2: This device is manufactured as a strain sensitive 4-terminal resistor. For compensation purposes, it is electrically modeled as a strain sensitive closed bridge.

APPLICATION GUIDE

Calibration and Temperature Compensation

The following compensation scheme can be used to calibrate and temperature compensate the basic sensor. It can be used in conjunction with amplification and other signal-conditioning circuitry.



Application Guide Notes:

Note 1: Temperature coefficient of sensitivity. To compensate, adjust R_1 in parallel with R_{TH} such that the temperature coefficient of the voltage across the sensor is of equal value but opposite in sign to the temperature coefficient of sensitivity.

Note 2: Full-Scale span (sensitivity). To adjust, vary R_S to achieve the desired value. An alternative approach for adjusting sensitivity is to use R_S as a shunt across the sensor output rather than putting R_S in series with the R_1 R_{TH} parallel combination as shown.

Note 3: Temperature coefficient of offset. Adjust by varying the thermistors R_{OTC1} and R_{OTC2} .

Note 4: Offset. Adjust using R_{01} and R_{02} .

PRESSURE SENSOR CHARACTERISTICS

Maximum Ratings

Supply voltage, V_S	$\leq V_{DC}$
Temperature Range	
Operating	-40°C to $+125^{\circ}\text{C}$
Storage	-55°C to $+155^{\circ}\text{C}$
Common-mode Line Pressure, SPX50D	$\approx 3\text{D}$
Lead Temperature	
Soldering, 10 seconds	200°C

Reference Conditions (Note 1)

Supply Voltage, V_S	$3V_{DC}$
Reference Temperature	25°C
Common-mode Line Pressure	0 psig

PERFORMANCE CHARACTERISTICS*

Characteristic	Min.	Typ.	Max.	Unit
Operating Pressure Range	0	—	50	kPa
Over Pressure	—	—	150	
Supply Voltage	—	3.0	6.0	V_{DC}
Supply Current	—	6.0	—	mA_{DC}
Full-scale Span (Note 2)	45	60	90	mV
Zero Pressure Offset	0	20	35	
Sensitivity	300	400	600	$\mu\text{V}/(\text{V} \cdot \text{kPa})$
Linearity and Hysteresis (Note 3)	—	± 0.10	± 0.50	%FSO
Repeatability (Note 4)	—	± 0.50	—	
Temperature Coefficient of Offset (Note 5)	—	± 15	—	$\mu\text{V}/^{\circ}\text{C}$
Temperature Coefficient of Sensitivity (Note 6)	-2200	-1900	-1600	$\text{ppm}/^{\circ}\text{C}$
Temperature Coefficient of Resistance (Note 7)	900	1100	1300	$\text{ppm}/^{\circ}\text{C}$
Input Resistance	400	—	600	Ω
Response Time (10% to 90%) (Note 8)	—	1.0	—	ms
Output impedance	—	600	—	Ω
Long Term Stability of Offset and Sensitivity (Note 9)	—	0.10	—	%FSO

*All specifications shown are "target" specifications. Subject to change.

Specification Notes:

Note 1: Conditions at which device Performance Characteristics apply.

Note 2: Full-scale is the algebraic difference between endpoints, where one endpoint is actual offset voltage and the other endpoint is the upper limit of the range (50 kPa).

Note 3: This is the algebraic sum of the percent of errors due to linearity and hysteresis given the following definitions:

$$\% \text{ FS error} = \frac{V_{1/2 \text{ full scale}} - \left\{ \left(\frac{V_{\text{full scale}} - V_{\text{offset}}}{\text{full scale pressure}} \right) \times \left(\frac{1}{2} \text{ full scale pressure} \right) + V_{\text{offset}} \right\}}{2 V_{\text{full scale}}} \times 100\%$$

(V = measured value for each device)

Hysteresis — the maximum output difference at any point within the operating pressure range for increasing and decreasing pressure.

Note 4: Maximum difference in output at any pressure with the operating pressure range and temperature within $+10^{\circ}\text{C}$ to -35°C after:

a) 1,000 temperature cycles, -40°C to $+125^{\circ}\text{C}$

b) 1.5 million pressure cycles, 0 kPa to 50 kPa

Note 5: Slope of the endpoint straight line fit to zero pressure offset at -40°C and $+125^{\circ}\text{C}$.

Note 6: Slope of the endpoint straight line fit to full-scale span at -40°C and $+125^{\circ}\text{C}$, relative to $+25^{\circ}\text{C}$.

Note 7: Slope of the endpoint straight line fit to input resistance at -40°C and $+125^{\circ}\text{C}$, relative to $+25^{\circ}\text{C}$.

Note 8: Response time for a 0 kPa to 50 kPa pressure step change.

Note 9: Long term stability is currently under test. Initial results show no measurable changes with time for offset or sensitivity.

Definitions:

kPa — Kilopascal 1 kPa equals 0.145 psi.

FSO — Full-Scale Output, same as Full-Scale Span.

LEI

PEJ

New Reference Tables
Supersedes N.B.S.
Circular #561

on Vs.
opper-
ckel
n-Constantan)

Temperature In Degrees C
ference Junction at 0°C

DEG C	0	1	2	3	4	5	6	7	8	9	10	DEG C
-210	-8.096											-210
-200	-7.870	-7.912	-7.934	-7.955	-7.976	-7.996	-8.017	-8.037	-8.057	-8.076	-8.096	-200
-190	-7.659	-7.683	-7.707	-7.731	-7.755	-7.778	-7.801	-7.824	-7.846	-7.868	-7.890	-190
-180	-7.402	-7.429	-7.455	-7.482	-7.508	-7.533	-7.559	-7.584	-7.609	-7.634	-7.659	-180
-170	-7.122	-7.151	-7.180	-7.209	-7.237	-7.265	-7.293	-7.321	-7.348	-7.375	-7.402	-170
-160	-6.821	-6.852	-6.883	-6.914	-6.944	-6.974	-7.004	-7.034	-7.064	-7.093	-7.122	-160
-150	-6.499	-6.532	-6.565	-6.598	-6.630	-6.663	-6.695	-6.727	-6.758	-6.790	-6.821	-150
-140	-6.159	-6.194	-6.228	-6.263	-6.297	-6.331	-6.365	-6.399	-6.433	-6.466	-6.499	-140
-130	-5.801	-5.837	-5.874	-5.910	-5.946	-5.982	-6.018	-6.053	-6.089	-6.124	-6.159	-130
-120	-5.426	-5.464	-5.502	-5.540	-5.578	-5.615	-5.653	-5.690	-5.727	-5.764	-5.801	-120
-110	-5.036	-5.076	-5.115	-5.155	-5.194	-5.233	-5.272	-5.311	-5.349	-5.388	-5.426	-110
-100	-4.632	-4.673	-4.714	-4.755	-4.795	-4.836	-4.876	-4.916	-4.956	-4.996	-5.036	-100
-90	-4.215	-4.257	-4.299	-4.341	-4.383	-4.425	-4.467	-4.508	-4.550	-4.591	-4.632	-90
-80	-3.785	-3.829	-3.872	-3.915	-3.958	-4.001	-4.044	-4.087	-4.130	-4.172	-4.215	-80
-70	-3.344	-3.389	-3.433	-3.478	-3.522	-3.566	-3.610	-3.654	-3.698	-3.742	-3.785	-70
-60	-2.892	-2.938	-2.984	-3.029	-3.074	-3.120	-3.165	-3.210	-3.255	-3.299	-3.344	-60
-50	-2.431	-2.478	-2.524	-2.570	-2.617	-2.663	-2.709	-2.755	-2.801	-2.847	-2.892	-50
-40	-1.960	-2.008	-2.055	-2.102	-2.150	-2.197	-2.244	-2.291	-2.338	-2.384	-2.431	-40
-30	-1.481	-1.530	-1.578	-1.626	-1.674	-1.722	-1.770	-1.818	-1.865	-1.913	-1.960	-30
-20	-0.995	-1.044	-1.093	-1.141	-1.190	-1.239	-1.288	-1.336	-1.385	-1.433	-1.481	-20
-10	-0.501	-0.550	-0.600	-0.650	-0.699	-0.748	-0.797	-0.847	-0.896	-0.945	-0.995	-10
0	0.000	0.050	0.101	0.151	0.202	0.253	0.303	0.354	0.405	0.456	0.507	0
10	0.507	0.558	0.609	0.660	0.711	0.762	0.813	0.865	0.916	0.967	1.019	10
20	1.019	1.070	1.122	1.174	1.225	1.277	1.329	1.381	1.432	1.484	1.536	20
30	1.536	1.588	1.640	1.693	1.745	1.797	1.849	1.901	1.954	2.006	2.058	30
40	2.058	2.111	2.163	2.216	2.268	2.321	2.374	2.426	2.479	2.532	2.585	40
50	2.585	2.638	2.691	2.743	2.796	2.849	2.902	2.954	3.007	3.060	3.113	50
60	3.113	3.166	3.221	3.275	3.328	3.381	3.435	3.488	3.542	3.595	3.649	60
70	3.649	3.702	3.756	3.809	3.863	3.917	3.971	4.024	4.078	4.132	4.186	70
80	4.186	4.239	4.293	4.347	4.401	4.455	4.509	4.563	4.617	4.671	4.725	80
90	4.725	4.780	4.834	4.888	4.942	4.996	5.050	5.105	5.159	5.213	5.268	90
100	5.268	5.322	5.376	5.431	5.485	5.540	5.594	5.649	5.703	5.758	5.812	100
110	5.812	5.867	5.921	5.976	6.031	6.085	6.140	6.195	6.249	6.304	6.359	110
120	6.359	6.414	6.468	6.523	6.578	6.633	6.688	6.742	6.797	6.852	6.907	120
130	6.907	6.962	7.017	7.072	7.127	7.182	7.237	7.292	7.347	7.402	7.457	130
140	7.457	7.512	7.567	7.622	7.677	7.732	7.787	7.842	7.897	7.952	8.008	140
150	8.008	8.063	8.118	8.174	8.229	8.284	8.339	8.394	8.449	8.505	8.560	150
160	8.560	8.616	8.671	8.726	8.781	8.837	8.892	8.947	9.003	9.058	9.113	160
170	9.113	9.169	9.224	9.279	9.335	9.390	9.446	9.501	9.556	9.612	9.667	170
180	9.667	9.723	9.778	9.834	9.889	9.944	10.000	10.055	10.111	10.166	10.222	180
190	10.222	10.277	10.333	10.388	10.444	10.499	10.555	10.610	10.666	10.721	10.777	190
200	10.777	10.832	10.888	10.943	10.999	11.054	11.110	11.165	11.221	11.276	11.332	200
210	11.332	11.387	11.443	11.498	11.554	11.609	11.665	11.720	11.776	11.831	11.887	210
220	11.887	11.943	11.998	12.054	12.109	12.165	12.220	12.276	12.331	12.387	12.442	220
230	12.442	12.498	12.553	12.609	12.664	12.720	12.776	12.831	12.887	12.942	12.998	230
240	12.998	13.053	13.109	13.164	13.220	13.275	13.331	13.386	13.442	13.497	13.553	240
250	13.553	13.608	13.664	13.719	13.775	13.830	13.886	13.941	13.997	14.052	14.108	250
260	14.108	14.163	14.219	14.274	14.330	14.385	14.441	14.496	14.552	14.607	14.663	260
270	14.663	14.718	14.774	14.829	14.885	14.940	14.995	15.051	15.106	15.162	15.217	270
280	15.217	15.273	15.328	15.383	15.439	15.494	15.550	15.605	15.661	15.716	15.771	280
290	15.771	15.827	15.882	15.938	15.993	16.048	16.104	16.159	16.214	16.270	16.325	290
300	16.325	16.380	16.436	16.491	16.547	16.602	16.657	16.713	16.768	16.823	16.879	300
310	16.879	16.934	16.989	17.044	17.100	17.155	17.210	17.266	17.321	17.377	17.432	310
320	17.432	17.487	17.542	17.597	17.653	17.708	17.763	17.818	17.874	17.929	17.984	320
330	17.984	18.039	18.095	18.150	18.205	18.260	18.316	18.371	18.426	18.481	18.537	330
340	18.537	18.592	18.647	18.702	18.757	18.813	18.868	18.923	18.978	19.033	19.089	340
350	19.089	19.144	19.199	19.254	19.309	19.364	19.420	19.475	19.530	19.585	19.640	350
360	19.640	19.695	19.751	19.806	19.861	19.916	19.971	20.026	20.081	20.137	20.192	360
370	20.192	20.247	20.302	20.357	20.412	20.467	20.522	20.578	20.633	20.688	20.743	370
380	20.743	20.798	20.853	20.909	20.964	21.019	21.074	21.129	21.184	21.239	21.294	380
390	21.294	21.350	21.405	21.460	21.515	21.570	21.625	21.680	21.735	21.791	21.846	390
400	21.846	21.901	21.956	22.011	22.066	22.122	22.177	22.232	22.287	22.342	22.397	400
410	22.397	22.453	22.508	22.563	22.618	22.673	22.728	22.783	22.838	22.893	22.948	410
420	22.948	23.003	23.058	23.113	23.168	23.223	23.278	23.333	23.388	23.443	23.500	420
430	23.501	23.556	23.611	23.667	23.722	23.777	23.832	23.887	23.942	23.997	24.053	430
440	24.054	24.109	24.164	24.220	24.275	24.330	24.386	24.441	24.496	24.552	24.607	440
450	24.607	24.662	24.718	24.773	24.829	24.884	24.939	24.995	25.050	25.106	25.161	450
460	25.161	25.217	25.272	25.327	25.383	25.438	25.493	25.549	25.604	25.660	25.715	460
470	25.716	25.772	25.827	25.883	25.938	25.994	26.050	26.105	26.161	26.216	26.272	470
480	26.272	26.328	26.383	26.439	26.495	26.551	26.606	26.662	26.718	26.773	26.829	480
490	26.829	26.885	26.941	26.997	27.053	27.109	27.165	27.220	27.276	27.332	27.388	490
500	27.388	27.444	27.500	27.556	27.612	27.668	27.724	27.780	27.836	27.892	27.949	500
510	27.949	28.005	28.061	28.117	28.173	28.230	28.286	28.342	28.398	28.455	28.511	510
520	28.511	28.567	28.624	28.680	28.736	28.793	28.849	28.906	28.962	29.019	29.075	520
530	29.075	29.132	29.188	29.245	29.301	29.358	29.415	29.471	29.528	29.585	29.642	530
540	29.642	29.698	29.755	29.812	29.869	29.926	29.983	30.039	30.096	30.153	30.210	540
550	30.210	30.267	30.324	30.381	30.439	30.496	30.553	30.610	30.667	30.724	30.782	550
560	30.782	30.839	30.896	30.954	31.011	31.068	31.126	31.183	31.241	31.298	31.356	560
570	31.356	31.413	31.471	31.528	31.586	31.644	31.702	31.759	31.817	31.875	31.933	570
580	31.933	31.991	32.048	32.106	32.164	32.222	32.280					

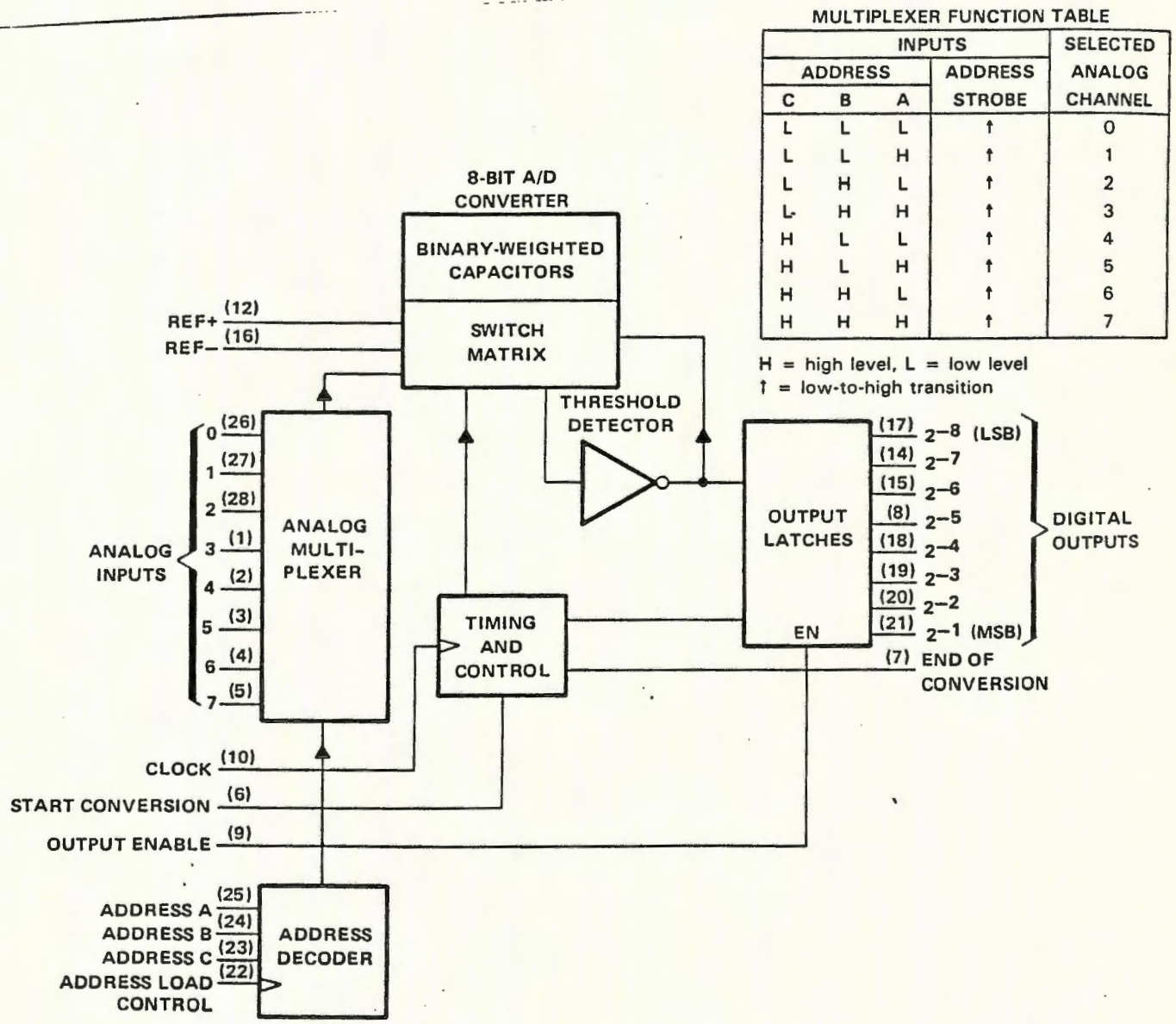


Figure 11-42. Functional Block Diagram (Positive) and Function Table

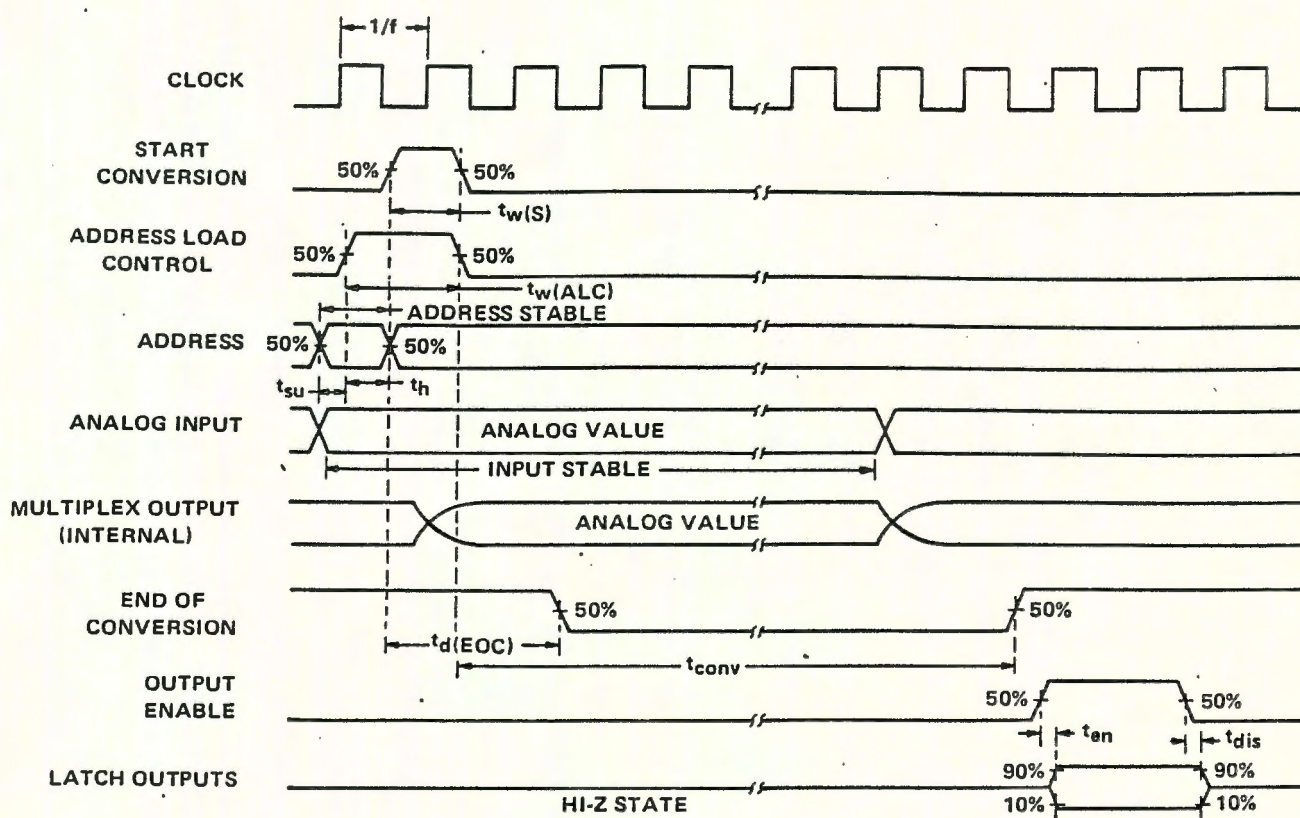


Figure 11-43. Operating Sequence

**SN54LS240, SN54LS241, SN54LS244, SN54S240, SN54S241, SN54S244,
SN74LS240, SN74LS241, SN74LS244, SN74S240, SN74S241, SN74S244**
OCTAL BUFFERS AND LINE DRIVERS WITH 3-STATE OUTPUTS

APRIL 1986—REVISED MARCH 1988

- 3-State Outputs Drive Bus Lines or Buffer Memory Address Registers
- PNP Inputs Reduce D-C Loading
- Hysteresis at Inputs Improves Noise Margins

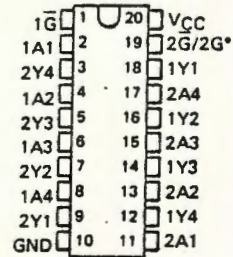
description

These octal buffers and line drivers are designed specifically to improve both the performance and density of three-state memory address drivers, clock drivers, and bus-oriented receivers and transmitters. The designer has a choice of selected combinations of inverting and noninverting outputs, symmetrical \bar{G} (active-low output control) inputs, and complementary G and \bar{G} inputs. These devices feature high fan-out, improved fan-in, and 400-mV noise-margin. The SN74LS' and SN74S' can be used to drive terminated lines down to 133 ohms.

The SN54' family is characterized for operation over the full military temperature range of -55°C to 125°C . The SN74' family is characterized for operation from 0°C to 70°C .

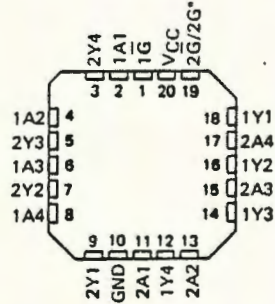
SN54LS', SN54S' ... J OR W PACKAGE
SN74LS', SN74S' ... DW OR N PACKAGE

(TOP VIEW)



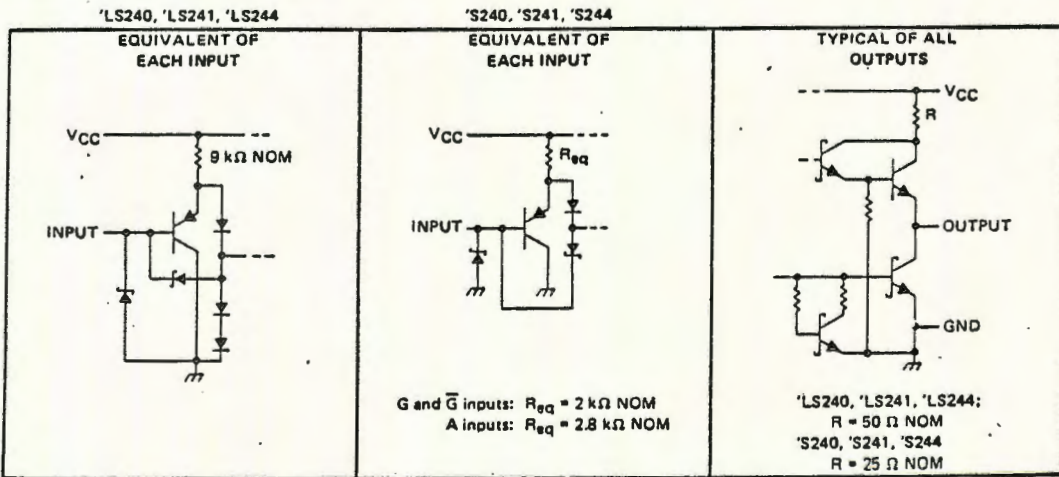
SN54LS', SN54S' ... FK PACKAGE

(TOP VIEW)



*2G for 'LS241 and 'S241 or 2G for all other drivers.

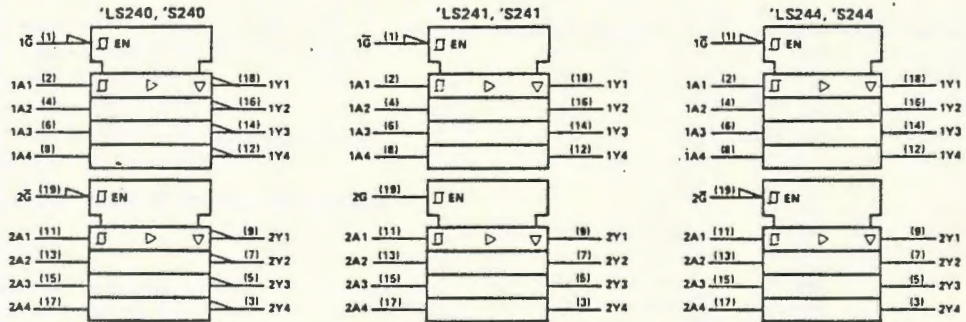
schematics of inputs and outputs



PRODUCTION DATA documents contain information 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.

**SN54LS240, SN54LS241, SN54LS244, SN54S240, SN54S241, SN54S244,
SN74SL240, SN74LS241, SN74LS244, SN74S240, SN74S241, SN74S244**
OCTAL BUFFERS AND LINE DRIVERS WITH 3-STATE OUTPUTS

logic symbols†

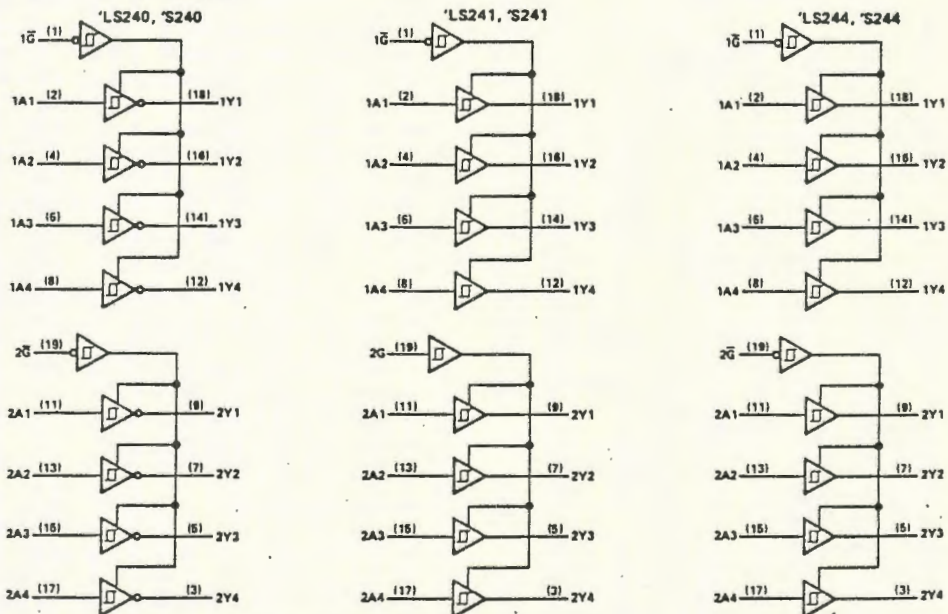


† These symbols are in accordance with ANSI/IEEE Std. 91-1984 and IEC Publication 617-12.

logic diagrams (positive logic)

2

TTL Devices



Pin numbers shown are for DW, J, N, and W packages.

absolute maximum ratings over operating free-air temperature range (unless otherwise noted)

Supply voltage, V_{CC} (see Note 1).....	7 V
Input voltage: 'LS Circuits.....	7 V
'S Circuits.....	5.5 V
Off-state output voltage.....	5.5 V
Operating free-air temperature range: SN54LS', SN54S' Circuits.....	-55°C to 125°C
SN74LS', SN74S' Circuits.....	0°C to 70°C
Storage temperature range.....	-65°C to 150°C

NOTE 1: Voltage values are with respect to network ground terminal.

SN54LS138, SN54S138, SN74LS138, SN74S138A 3-LINE TO 8-LINE DECODERS/DEMULPLEXERS

DECEMBER 1972 - REVISED MARCH 1988

- Designed Specifically for High-Speed: Memory Decoders Data Transmission Systems
- 3 Enable Inputs to Simplify Cascading and/or Data Reception
- Schottky-Clamped for High Performance

description

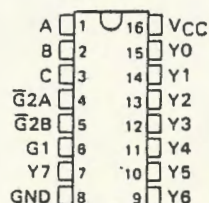
These Schottky-clamped TTL MSI circuits are designed to be used in high-performance memory decoding or data-routing applications requiring very short propagation delay times. In high-performance memory systems, these decoders can be used to minimize the effects of system decoding. When employed with high-speed memories utilizing a fast enable circuit, the delay times of these decoders and the enable time of the memory are usually less than the typical access time of the memory. This means that the effective system delay introduced by the Schottky-clamped system decoder is negligible.

The 'LS138, SN54S138, and SN74S138A decode one of eight lines dependent on the conditions at the three binary select inputs and the three enable inputs. Two active-low and one active-high enable inputs reduce the need for external gates or inverters when expanding. A 24-line decoder can be implemented without external inverters and a 32-line decoder requires only one inverter. An enable input can be used as a data input for demultiplexing applications.

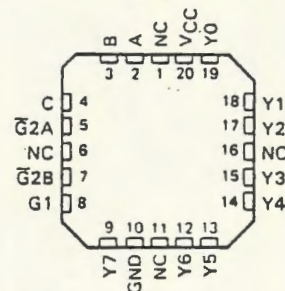
All of these decoder/demultiplexers feature fully buffered inputs, each of which represents only one normalized load to its driving circuit. All inputs are clamped with high-performance Schottky diodes to suppress line-ringing and to simplify system design.

The SN54LS138 and SN54S138 are characterized for operation over the full military temperature range of -55°C to 125°C . The SN74LS138 and SN74S138A are characterized for operation from 0°C to 70°C .

SN54LS138, SN54S138 . . . J OR W PACKAGE
SN74LS138, SN74S138A . . . D OR N PACKAGE
(TOP VIEW)

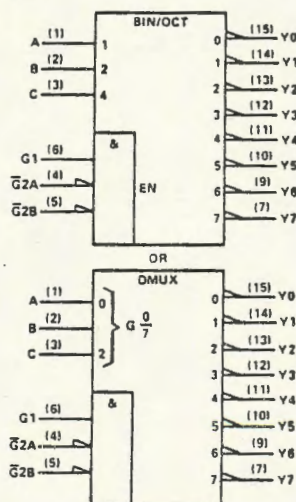


SN54LS138, SN54S138 . . . FK PACKAGE
(TOP VIEW)



NC—No internal connection

logic symbols†



†These symbols are in accordance with ANSI/IEEE Std 91-1984 and IEC Publication 617-12.

Pin numbers shown are for D, J, N, and W packages.

PRODUCTION DATA documents contain information 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.

TEXAS
INSTRUMENTS

POST OFFICE BOX 655012 • DALLAS, TEXAS 75265

Copyright © 1972, Texas Instruments Incorporated

2-425

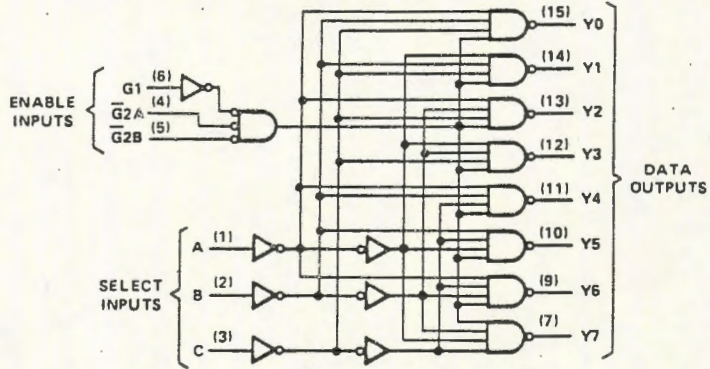
2

TTL Devices

SN54LS138, SN54S138, SN74LS138, SN74S138A
3-LINE-TO 8-LINE DECODERS/DEMULTIPLEXERS

logic diagram and function table

'LS138, SN54S138, SN74S138A



Pin numbers shown are for D, J, N, and W packages.

'LS138, SN54S138, SN74S138A
 FUNCTION TABLE

INPUTS					OUTPUTS							
ENABLE		SELECT			Y0	Y1	Y2	Y3	Y4	Y5	Y6	Y7
G1	$\bar{G}2^*$	C	B	A								
X	H	X	X	X	H	H	H	H	H	H	H	H
L	X	X	X	X	H	H	H	H	H	H	H	H
H	L	L	L	L	L	H	H	H	H	H	H	H
H	L	L	L	H	H	L	H	H	H	H	H	H
H	L	L	L	L	H	H	L	H	H	H	H	H
H	L	L	H	H	H	H	L	H	H	H	H	H
H	L	L	H	L	H	H	H	L	H	H	H	H
H	L	H	L	H	H	H	H	H	L	H	H	H
H	L	H	H	L	H	H	H	H	H	L	H	H
H	L	H	H	H	H	H	H	H	H	H	L	H
H	L	H	H	H	H	H	H	H	H	H	H	L

* $\bar{G}2 = \bar{G}2A + \bar{G}2B$
 H = high level, L = low level, X = irrelevant

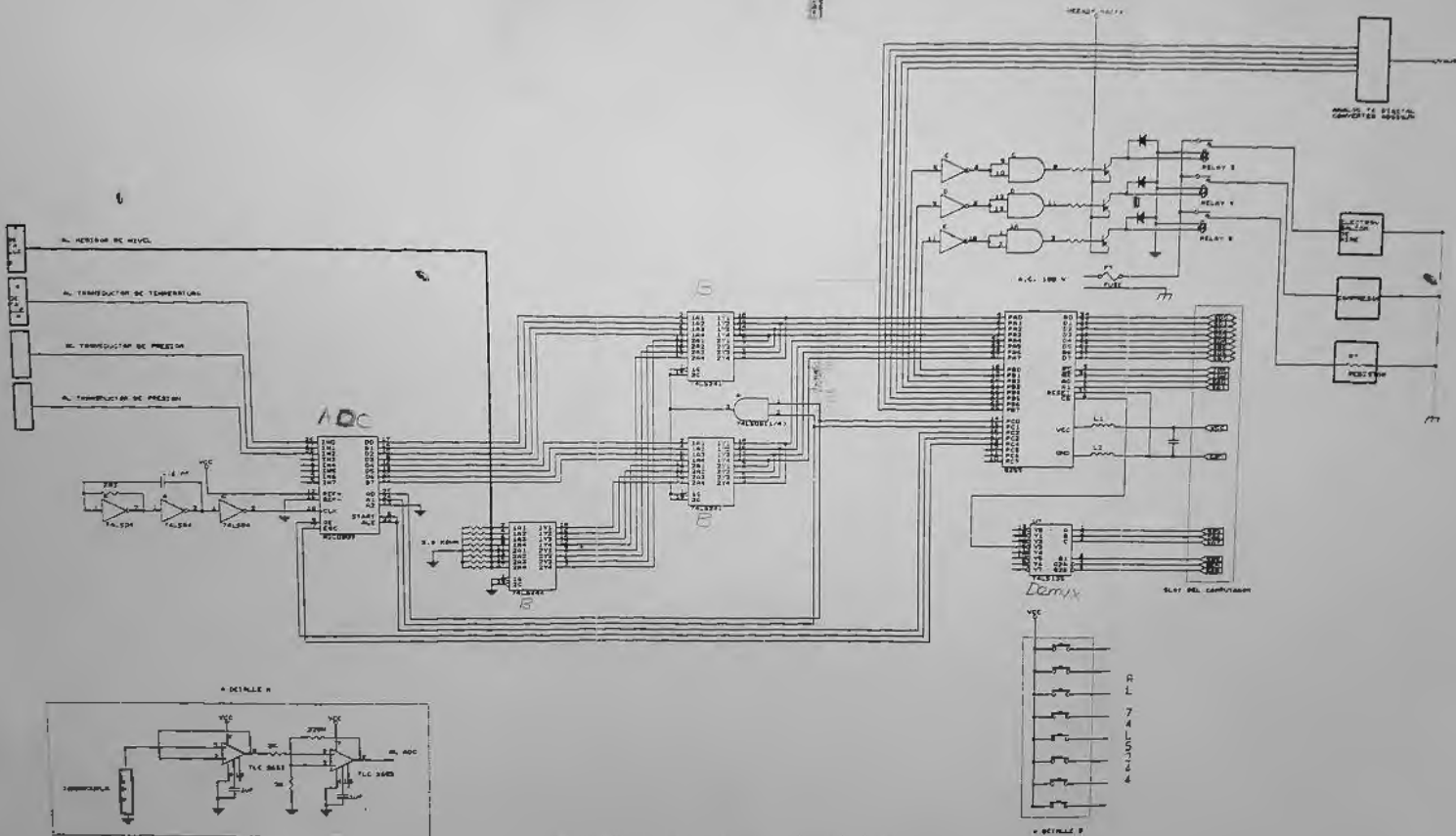
2 TTL Devices

ANEXO 7

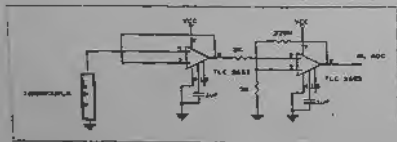
MODO 0 (MODO SIMPLE DE LOS PUERTOS COMO E/S)

16 CONFIGURACIONES POSIBLES

	PA	PB	PCL	PCU	D7	D6	D5	D4	D3	D2	D1	D0
	IN	IN	IN	IN	1	0	0	1	1	0	1	1
	IN	IN	IN	OUT	1	0	0	1	0	0	1	1
	IN	IN	OUT	IN	1	0	0	1	1	0	1	0
	IN	IN	OUT	OUT	1	0	0	1	0	0	1	0
	IN	OUT	IN	IN	1	0	0	1	1	0	0	1
	IN	OUT	IN	OUT	1	0	0	1	0	0	0	1
152	IN	OUT	OUT	IN	1	0	0	1	1	0	0	0
	IN	OUT	OUT	OUT	1	0	0	1	0	0	0	0
	OUT	IN	IN	IN	1	0	0	0	1	0	1	1
	OUT	IN	IN	OUT	1	0	0	0	0	0	1	1
	OUT	IN	OUT	IN	1	0	0	0	1	0	1	0
	OUT	IN	OUT	OUT	1	0	0	0	0	0	1	0
	OUT	OUT	IN	IN	1	0	0	0	1	0	0	1
	OUT	OUT	IN	OUT	1	0	0	0	0	0	0	1
	OUT	OUT	OUT	IN	1	0	0	0	1	0	0	0
	OUT	OUT	OUT	OUT	1	0	0	0	0	0	0	0



* DETALLE A



* DETALLE B



ANEXO B CIRCUITO PRINCIPAL

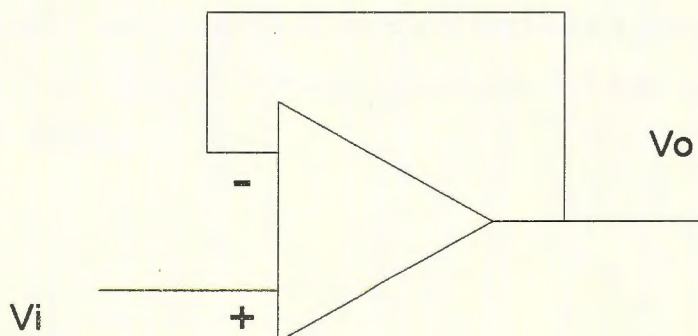
ANEXO 9**AMPLIFICADORES OPERACIONALES**

Las señales de la mayoría de los sensores son bastante pequeñas, así que estas deben ser amplificadas. Esto es usualmente hecho con algún tipo de amplificador operacional. Los amplificadores operacionales son dispositivos que tienen características especiales tales como: alta impedancia de entrada, alta ganancia, responde a la alta frecuencia, alta inmunidad al ruido, etc.

Las configuraciones utilizadas en este trabajo con amplificadores operacionales son las siguientes: seguidor de voltaje y amplificador no inversor.

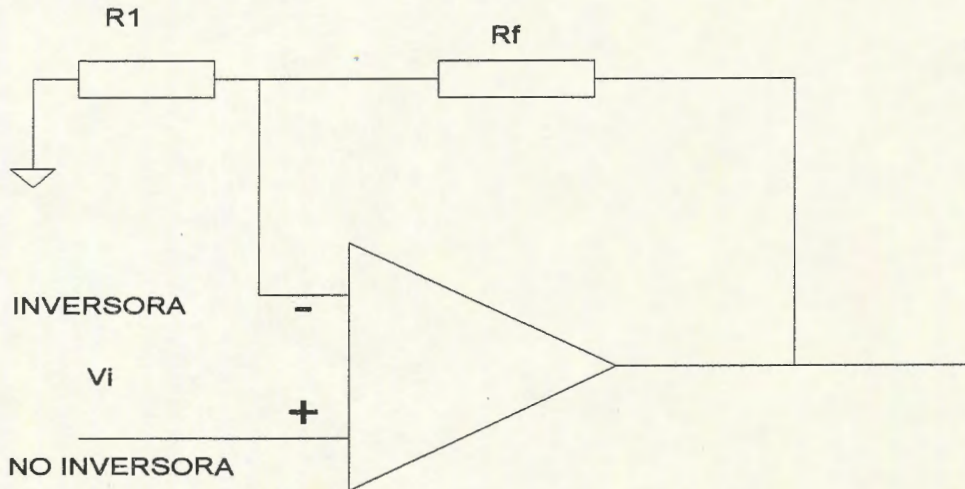
SEGUIDOR DE VOLTAJE

Esta configuración es utilizada en aplicaciones donde se requiere interconectar fuentes de alta impedancia y cargas de baja impedancia. Entre sus características especiales tenemos: alta impedancia de entrada y baja impedancia a la salida. La ganancia en esta configuración es de uno. Esta configuración de seguidor de voltaje es un caso especial de amplificador no inversor donde todo el voltaje de salida es retroalimentado a la entrada inversora a través de la conexión de retroalimentación con ganancia de uno. El seguidor de voltaje funciona extrayendo una corriente despreciable de la fuente de señal, esta señal encuentra en el amplificador operacional un circuito abierto. Por lo tanto el voltaje de entrada es igual al voltaje de salida. Esta configuración se muestra en la siguiente figura:



AMPLIFICADOR NO INVERSOR

En esta configuración el voltaje a la salida mantiene la misma polaridad que el voltaje a la entrada. La siguiente figura muestra esta configuración:



El funcionamiento de este circuito es el siguiente: para fines prácticos se considera como cero el voltaje entre los terminales no inversor e inversor, por lo tanto el voltaje aplicado en la terminal no inversor e inversor se encuentran al mismo potencial, de allí que el voltaje de entrada aparece en la resistencia R_1 lo cual causa que la corriente fluya hacia tierra. La corriente fluye a través de R_f y su caída de voltaje representa V_{Rf} y se expresa como $V_{Rf} = (R_f/R_1)V_i$, entonces el voltaje de salida $V_o = V_i + V_{Rf}$, de aquí se puede expresar que la ganancia de voltaje (V_o/V_i) esta dada por $(1 + R_f/R_1)$. Debido a este comportamiento se efectúa la amplificación. Cabe notar que la corriente en la carga que se conecte al amplificador operacional esta dada por V_o/R_l , de allí que la I_l solo depende de V_o que es generado de acuerdo a la ganancia que se desea y de R_l la resistencia de carga.

BIBLIOGRAFIA**8086/8088 MICROPROCESSOR: Architecture Programming and Interfacing**

Barry B. Brey

Editorial MERRIL

ELECTRONIC DESIGN: Circuits and Design

Savant/Rodent/Carpenter

Editorial: Benjamin Cummins

LINEAR & INTERFACE CIRCUITS APPLICATIONS

Texas Instrument Engineering Staff

Editorial: McGrawHill

MECANICA DE FLUIDOS

V́ctor L. Streeter

Editorial: Mc Graw Hill

SISTEMAS DIGITALES : Principios y Aplicaciones

Ronald J. Tocci

Editorial: PRENTICE HALL

VISUAL BASIC :Programmers guide to the Windows API

Daniel Appleman

Editorial : Ziff Davis Press

THE VISUAL BASIC 3 FOR WINDOWS HANDBOOK

Gary Cornell

Editorial: OSBORNE

VISUAL BASIC FOR WINDOWS

Richard Mansfield

Editorial: VENTANA PRESS

WINDOWS PROGRAMMING: An Introduction

William Murray/Chris Papas

Editorial: OSBORNE/McGrawHill