



**UNIVERSIDAD NACIONAL AUTONOMA
DE MEXICO**

FACULTAD DE INGENIERIA

DISEÑO Y CONSTRUCCION
DE UNA BANDA TRANSPORTADORA

T E S I S

QUE PARA OBTENER EL TITULO DE:
INGENIERO MECANICO ELECTRICISTA
AREA: ELECTRICO - ELECTRONICO

P R E S E N T A :
JORGE ENRIQUE WALS SELVAS

DIRECTOR DE TESIS: ING. FRANCISCO RODRIGUEZ RAMIREZ



**TESIS CON
FALLA DE ORIGEN**

MEXICO, D.F.

2002



Universidad Nacional
Autónoma de México

Dirección General de Bibliotecas de la UNAM

Biblioteca Central



UNAM – Dirección General de Bibliotecas
Tesis Digitales
Restricciones de uso

DERECHOS RESERVADOS ©
PROHIBIDA SU REPRODUCCIÓN TOTAL O PARCIAL

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

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

PAGINACIÓN

DISCONTINUA



FACULTAD DE INGENIERIA
DIVISION DE INGENIERIA ELECTRICA

OFICIO FING/DIE/325/2002

ASUNTO: Solicitud de jurado para
Examen Profesional

M.C. GERARDO JOSÉ FERRANDO BRAVO
DIRECTOR DE LA FACULTAD DE
INGENIERÍA DE LA U.N.A.M.
Presente.

A través de la Coordinación de Seminarios, el Señor WALS SELVAS JORGE ENRIQUE con número de cuenta 8935163-5 de la carrera INGENIERO MECÁNICO ELECTRICISTA AREA: ELÉCTRICO ELECTRÓNICO; habiendo satisfecho los requisitos académicos necesarios para efectuar su examen profesional, le solicita atentamente autorizarle, tanto la fecha:

06 DE MAYO DEL 2002 A LAS 9:00 HORAS

así como el siguiente jurado

| | |
|-------------|----------------------------------|
| PRESIDENTE: | ING. RODOLFO PETERS LAMMEL |
| VOCAL: | ING. FRANCISCO RODRÍGUEZ RAMÍREZ |
| SECRETARIO: | ING. SABINO ORTEGA MONJARÁS |
| 1ER. SPTE.: | ING. SILVINA HERNÁNDEZ GARCÍA |
| 2DO. SPTE.: | ING. JOSÉ LUIS RODRÍGUEZ PÉREZ |

Para cualquier aclaración o duda al respecto, favor de comunicarse a la Coordinación de Seminarios y Servicio Social con el Ing. José Arturo Origel Coutiño, número telefónico 5622-3111.

Atentamente.
"POR MI RAZA HABLARA EL ESPÍRITU"
Cd. Universitaria, D. F., a 29 de Abril del 2002

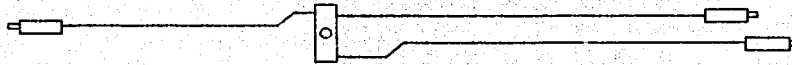
EL JEFE DE LA DIVISIÓN


M.I. LUIS ARTURO HARO RUIZ


FIRMA DEL ALUMNO

NOTA:

- 1.- Es responsabilidad del alumno recordar a cada uno de sus sinodales la fecha y hora del examen un día antes de su realización.
- 2.- En caso de que se modifique la fecha y/u hora de examen señalada en el presente, oficio, el alumno deberá contar nuevamente con la aprobación de todos sus sinodales.



A:

DIOS PADRE POR ESTAR SIEMPRE CONMIGO.

MI ABUELO POR SEMBRAR EN MI EL DESEO DE APRENDER
MI MADRE QUE SIEMPRE ME HA DEJADO SER
MI HERMANA LA MEJOR ADMIRADORA QUE TENGO.

A PACO POR SU PACIENCIA Y APOYO ICONDICIONAL
MIS PROFESORES A TODOS ELLOS, CON AFECTO POR COMPARTIR SU SABER.

MI MADRE "PSIQUE" QUIEN IMPUSLO MI NACIMIENTO SIMBOLICO
MIS MENTORES ADULTOS, CUYOS CONSEJOS ME HAN GUIADO

MIS AMGOS POR CREER QUE YO PODRIA "INVENTAR ...UN POPOTE"

A TODA LA GRAN FAMILIA CON QUE HOY EN DIA CUENTO SIN RESERVAS
MESSOA (✚) MI HERMOSO ANGEL DE LA GUARDIA
MI PAREJA POR SU AMOR
XL ()
Y ¿POR QUÉ NO? A MI VIEJO SANSON

A TODOS GRACIAS DE CORAZON
JORGE E. WALS



ÍNDICE TEMÁTICO.

CAPITULO

PAGINA

INTRODUCCIÓN

- Planteamiento del problema.....I.1
- Objetivo del sistema.....I.1
- Alcances del sistema a desarrollar.....I.1
- Utilización del sistema.....I.2
- Descripción del trabajo de tesis.....I.2

1. - SENSORES.

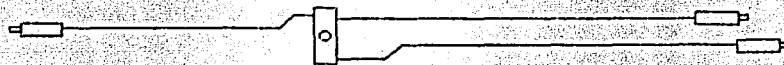
- 1.1. - Criterio de selección. ¿Por qué un sensor fotoeléctrico?.....1.1
 - 1.1.1. - Ventajas de un sensor óptico.....1.1
 - 1.1.2. - Ventajas del manejo de señales eléctricas.....1.1
- 1.2. - Definición de transductores y sensores.....1.2
- 1.3. - Tipos de sensores.....1.3
- 1.4. - Características de los sensores.....1.6
 - 1.4.1. - Exactitud, fidelidad, sensibilidad.....1.6
 - 1.4.2. - Linealidad, resolución.....1.7
- 1.5. - Forma de conexión de los sensores implementados.....1.9

2. - LENGUAJES DE ALTO NIVEL

- 2.1. - Clasificación de los lenguajes de programación.....2.1
- 2.2. - Construcciones para el manejo de estructuras de control.....2.3
 - 2.2.1. - Construcciones para el control de ciclos.....2.3
 - 2.2.2. - Construcciones de decisión.....2.4
 - 2.2.3. - Control de excepciones en un programa.....2.4
- 2.3. - Ventajas del lenguaje Visual Basic 3.0 Profesional Para Windows.....2.4
- 2.4. - Programa manejador de la banda transportadora.....2.5

3. - MICROCONTROLADORES Y CIRCUITOS DIGITALES.

- 3.1. - Definición de la lógica binaria.....3.1
- 3.2. - Circuitos digitales combinacionales.....3.2
 - 3.2.1. - Álgebra booleana.....3.2
 - 3.2.2. - Compuertas lógicas.....3.3
- 3.3. - Lógica combinatorial con MSI y LSI.....3.6
 - 3.3.1. - Decodificadores.....3.6
 - 3.3.2. - Demultiplexores.....3.8
 - 3.3.3. - Codificadores.....3.9
 - 3.3.4. - Multiplexores.....3.10
 - 3.3.5. - Memoria de solo lectura ROM.....3.11
 - 3.3.6. - Arreglo lógico programable PLA.....3.13
- 3.4. - Microcontroladores.....3.15



| | |
|---|------|
| 3.5. - Empleo del microcontrolador 87C51 de Intel..... | 3.15 |
| 3.5.1. - Programa y Diagrama de Flujo del Microcontrolador..... | 3.17 |
| 4. - PUERTOS DE UNA COMPUTADORA PERSONAL E INTERFAZ | |
| 4.1. - Puertos típicos de una computadora personal..... | 4.1 |
| 4.2. - Modo de operación de un puerto paralelo..... | 4.1 |
| 4.3. - Modo de operación de un puerto serie..... | 4.2 |
| 4.4. - Aplicación del puerto serie con una interfaz RS - 232..... | 4.2 |
| 5. - DISTRIBUCIÓN DE LA BANDA TRANSPORTADORA. | |
| 5.1. - Especificaciones principales de la banda transportadora..... | 5.1 |
| 5.2. - Diagrama de la banda transportadora..... | 5.2 |
| 5.3. - Ubicación de los sensores en la banda transportadora..... | 5.2 |
| 5.4. - Circuito manejador del Motor de corriente alterna..... | 5.3 |
| 6. - BASE DE DATOS | |
| 6.1. - Conceptos básicos..... | 6.1 |
| 6.2. - Sistemas manejadores de bases de datos..... | 6.2 |
| 6.3. - Organización de la base de datos implementada..... | 6.3 |
| 7. - ANÁLISIS ESTADÍSTICO. | |
| 7.1. - Definición de estadística..... | 7.1 |
| 7.2. - Elementos de estadística..... | 7.1 |
| 7.2.1. - Datos no agrupados..... | 7.1 |
| 7.2.2. - Datos agrupados..... | 7.4 |
| 7.3. - Medidas de tendencia central..... | 7.7 |
| 7.3.1. - La media..... | 7.7 |
| 7.3.2. - La mediana..... | 7.8 |
| 7.3.3. - La moda..... | 7.9 |
| 7.4. - Medidas de Dispersión..... | 7.9 |
| 7.4.1. - Rango..... | 7.10 |
| 7.4.2. - Desviación estándar..... | 7.10 |
| 8. - CARACTERÍSTICAS DE CALIDAD. | |
| 8.1. - Introducción del término "Calidad"..... | 8.1 |
| 8.2. - Variables y Atributos de calidad..... | 8.1 |
| 8.3. - Control Estadístico del Proceso..... | 8.2 |
| 8.3.1. - Método de la gráfica de control..... | 8.4 |
| 8.4. - Interpretación de datos..... | 8.5 |
| 8.4.1. - Concepto de población y muestra..... | 8.5 |



9. - CONCLUSIONES.

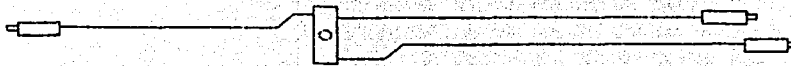
9.1.- Revisión de puntos originales.....9.1

9.2.- Expansiones a futuro para el prototipo.....9.2

APÉNDICE A

- Fuente de alimentación.....A.1

BIBLIOGRAFÍA.....B.1



DISEÑO Y CONSTRUCCIÓN DE UNA BANDA TRANSPORTADORA.

INTRODUCCIÓN

- Planteamiento del problema.

Para efectos didácticos en el área de ingeniería industrial se requiere de un instrumento de practica. Que simulara un proceso industrial donde se sensarán algunas variables de calidad, así mismo de manera automatizada el proceso deberá ser controlado por una computadora personal. También se requerirá que las variables sensadas, se obtengan como resultado en una base de datos, compatible con software comercial.

- Objetivo del sistema.

Se desarrollará un sistema electrónico, que simule en pequeña escala, una banda transportadora de un proceso industrial, y que a través de sensores capte las dimensiones de las piezas del proceso.

Además el sistema contará con una adecuada interfaz que envíe y guarde la información en un archivo para una base de datos compatible con software comercial, evitando así la captura manual y controlando todo el proceso a través de una computadora personal.

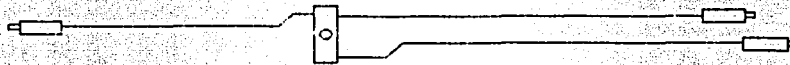
- Alcances del sistema a desarrollar.

Para alcanzar el objetivo del sistema se implementarán tres sensores fotoeléctricos, cada uno medirá una dimensión de la pieza en cuestión (largo, ancho y profundidad) estos enviarán su información a unos circuitos digitales coordinados con un microcontrolador.

El microcontrolador trabajará con su propia señal de reloj, de manera que el sistema se encuentre sincronizado y pueda interpretar el resultado de las mediciones de los sensores.

La información de los sensores, se enviará a la interfaz del sistema, esta será conectada a cualquier computadora personal, a través de su puerto serie.

La pieza a medirse será colocada en una banda transportadora. Sobre la cual estarán montados los sensores. El inicio y paro del avance y medición de la pieza sobre la banda será controlado por la misma computadora personal.



Los resultados obtenidos por el proceso se guardarán en la computadora en un archivo, que podrá ser abierto y modificado posteriormente, con software comercial.

- Utilización del sistema.

El sistema será empleado con fines puramente didácticos, para reforzar el aprendizaje de los alumnos de la área de ingeniería industrial, simulándoles un proceso real. Para dicho efecto se desarrolló el prototipo que en estas páginas se explica. Este mismo prototipo fue diseñado desde el inicio bajo la premisa de ser flexible y modular. Esto significa que en un corto plazo alumnos de diversas áreas podrán participar en el desarrollo de nuevas expansiones o mejoras para el mismo. Con el afán de hacer accesible el desarrollo de nuevas tecnologías para los alumnos de esta facultad, se encuentran en el presente trabajo todos los códigos fuente y diagramas que componen el prototipo.

Cabe mencionar que el mismo será donado al laboratorio de pruebas y calidad, para que se le dé el uso para que fuese destinado desde su fase de diseño.

- Descripción del trabajo de tesis.

Con el fin de conocer la estructura que conforma el presente trabajo, a continuación se hará una breve descripción sobre el contenido de cada capítulo y las ideas principales que lo componen.

Los capítulos comprendidos del número 1 al número 6 tienen en su desarrollo tanto la parte teórica como la parte práctica. La teoría se refiere a los fundamentos básicos del capítulo en cuestión. En tanto que, la práctica es la documentación del diseño obtenido para este trabajo. En estos casos se describirán los puntos fundamentales a describir de teoría y/o de práctica según sea el caso.

El capítulo 1 "Sensores", se refiere a los sensores ópticos del sistema, en este capítulo la teoría nos define los principales parámetros de un sensor y el por qué seleccionamos los del tipo óptico, la práctica nos identifica de cuántas etapas se compone cada sensor y qué función tienen. También en el capítulo 1 podemos ver el diagrama general de los componentes de cada sensor.

El capítulo 2 "Lenguajes de alto nivel", nos describe la clasificación de los lenguajes de programación y también la construcción de manejo de estructuras de control en un programa. En el mismo capítulo se describen las principales razones para utilizar un lenguaje de programación visual y se incluye también el código fuente del programa ejecutable, que opera la banda transportadora.



En el capítulo 3 "Microcontroladores y circuitos digitales", se presentan los conceptos básicos de la lógica binaria, el uso de compuertas y sus correspondientes tablas de verdad. Se mencionan también los diferentes dispositivos para emplear en un diseño digital, desde las mencionadas compuertas hasta llegar a los microcontroladores, ahí mismo se explican las ventajas que presenta emplear un microcontrolador como parte del diseño.

En el capítulo 4 "Puertos de una computadora personal e interfaz", se hace una breve descripción de los puertos más comunes en una computadora personal y su modo de funcionamiento. También se indica la manera de utilizar un puerto serie con interfaz RS-232, para intercomunicar el puerto serial con un periférico externo.

El capítulo 5 "Distribución de la banda transportadora", describe las dimensiones y especificaciones de los materiales que componen a la banda transportadora. También se indica la ubicación física de los sensores en la banda y se muestra el diagrama del circuito manejador del motor de corriente alterna.

El capítulo 6 "Base de datos", se refiere a los conceptos actuales que conforman una base de datos y como aplican a este tema de tesis. También se muestra un ejemplo de una base de datos obtenida con el prototipo.

Los capítulos 7 y 8 son teóricos únicamente por lo que la descripción practica en estos casos no aplica.

El capítulo 7 "Análisis estadístico" nos presenta los principales conceptos de estadística, con datos agrupados y no agrupados, medidas de tendencia central, así como medidas de dispersión.

El capítulo 8 "Características de calidad" nos acerca a los principales parámetros de calidad como son: variables y atributos, control estadístico del proceso y los principales conceptos utilizados en la interpretación de datos, que son los conceptos de población y muestra.

El capítulo 9 "Conclusiones" tiene como único objetivo resumir como se consiguieron los objetivos del sistema, tanto de manera particular, como de manera general.

Finalmente como tópico adicional, en el "Apéndice A" se presenta el diagrama de la fuente de alimentación del sistema y la disponibilidad que tiene para las futuras expansiones del prototipo.



1. - SENSORES.

1.1. - Criterio de selección, ¿ Por qué un sensor fotoeléctrico?

1.1.1. - Ventajas de un sensor óptico.

Dado que hay seis tipos de señales: mecánicas, magnéticas, térmicas, eléctricas, ópticas y moleculares (química), en esta sección trataremos las señales que se utilizarán en nuestro caso.

La descripción de los diferentes tipos de sensores se describirá de manera general en la sección 1.3.

Para nuestros requerimientos optamos por un sensor que no tuviese contacto físico con la pieza a medir, evitando así problemas mecánicos o mal funcionamiento por desgaste en un plazo a término medio.

Un sensor fotoeléctrico tiene las siguientes ventajas respecto a otros sensores comerciales:

1. - Mantenimiento prácticamente nulo, (basta limpiarlo eventualmente)
2. - Es mucho más barato.
3. - Su diseño, montaje y utilización, es relativamente simple.

1.1.2. - Ventajas del manejo de señales eléctricas.

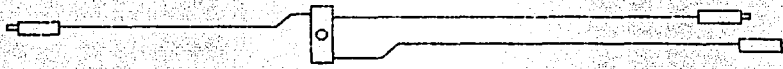
Un sistema de medida electrónico ofrece entre otras las siguientes ventajas:

1. - Debido a la estructura electrónica de la materia, cualquier variación de un parámetro no eléctrico de un material viene acompañada por la variación de un parámetro eléctrico.

Esto permite realizar transducciones con salida eléctrica para cualquier magnitud física no eléctrica.

2. - Dado que en el proceso de medida no conviene extraer energía del sistema donde se mide, lo mejor es amplificar la señal de salida del transductor. Con amplificadores electrónicos se pueden obtener fácilmente ganancias de potencia de 10^{10} .

3. - Además de la amplificación, hay una gran variedad de recursos, en forma de circuitos integrados, para acondicionar o modificar las señales eléctricas. Incluso hay transductores que incorporan físicamente en un mismo encapsulado parte de estos recursos.



4. - Existen también numerosos recursos para presentar o registrar información si se hace electrónicamente, pudiéndose manejar no solo datos numéricos, sino también textos, gráficos y diagramas.

5. - La transmisión de señales eléctricas es más versátil que la de señales mecánicas, hidráulicas o neumáticas.

Conjuntando las ventajas del manejo de una señal eléctrica y las opciones favorables de un sensor óptico, se opta por el empleo de la unión de ambos beneficios, a través de sensores fotoeléctricos.

1.2. - Definición de transductores y sensores.

Transductor.- Dispositivo que convierte una magnitud física en otra distinta por ejemplo conversión de temperatura en tensión eléctrica.

Esto significa que la señal de entrada es siempre una energía o potencia.

Al medir, una de las componentes de la señal suele ser tan pequeña que puede despreciarse, y se interpreta que se mide solo la otra componente.

Al medir una fuerza, por ejemplo, se supone que el desplazamiento del transductor es despreciable, es decir que no se "carga" al sistema, ya que de lo contrario podría suceder que este fuera incapaz de aportar la energía necesaria para el desplazamiento. En la transducción siempre se extrae una cierta energía del sistema donde se mide, por lo que es importante garantizar que esto no lo perturba.

Cualquier dispositivo que convierta una señal de un tipo en una señal distinta debería considerarse un transductor, y la señal de salida podría ser de cualquier forma "útil". En la práctica no obstante se consideran transductores en general, aquellos que ofrecen una señal de salida eléctrica.

Sensor.- Dispositivo que detecta variaciones en una magnitud física y las convierte en señales útiles para un sistema de medida o control.

Sensor y transductor se emplean a veces como sinónimos, sin embargo, sensor sugiere un significado más extenso: la ampliación de los sentidos para adquirir un conocimiento de cantidades físicas que, por su naturaleza o tamaño, no pueden ser percibidas directamente por los sentidos.

Para efectos de este trabajo, se utilizará el término sensor para referirse a un transductor de entrada. No se tratan los accionamientos o transductores de salida.



A veces, sobre todo en el caso de magnitudes mecánicas, puede señalarse la presencia de un elemento designado como sensor primario, que convierte la variable de medida, en una señal de medida, siendo el sensor electrónico quien la convierte en una señal eléctrica.

1.3. - Tipos de sensores.

El número de sensores disponibles para las distintas magnitudes físicas es tan elevado que no se puede proceder racionalmente a su estudio sin clasificarlos previamente de acuerdo con algún criterio.

Según el aporte de energía, los sensores se pueden dividir en moduladores y generadores.

Según la señal de salida los sensores se clasifican en analógicos o digitales. En los analógicos la señal varía de forma continua. La información está en la amplitud, si bien suelen incluir en este grupo los sensores con salida en el dominio temporal.

En los sensores discretos la salida varía en forma de saltos o pasos discretos. No requieren conversión A/D y la transmisión de su salida es más fácil. Tienen también mayor fidelidad y fiabilidad, y muchas veces mayor exactitud, pero lamentablemente no hay modelos digitales para muchas de las magnitudes físicas de mayor interés.

Atendiendo al modo de funcionamiento, los sensores pueden ser de deflexión o de comparación.

En los sensores que funcionan por deflexión, la magnitud medida produce algún efecto físico, que engendra algún efecto similar, sin embargo, es opuesto, en alguna parte del instrumento, ya que está relacionado con alguna variable útil.

Un dinamómetro para la medida de fuerzas es un sensor de este tipo en el que la fuerza aplicada deforma un muelle hasta que la fuerza de recuperación de éste es proporcional a su longitud, e igual a la fuerza aplicada.

En los sensores que funcionan por comparación, se intenta mantener nula la deflexión mediante la aplicación de un efecto bien conocido, opuesto al generado por la magnitud a medir. Hay un detector del desequilibrio y un medio para establecerlo. En una balanza manual, por ejemplo, la colocación de una masa en un platillo provoca un desequilibrio, indicado por una aguja sobre una escala. El operario coloca entonces una o varias masas en el otro platillo hasta alcanzar el equilibrio, que se juzga por la posición de la aguja.



Las medidas por comparación suelen ser más exactas porque el efecto conocido opuesto se puede calibrar con un patrón o magnitud de referencia de calidad.

El detector de desequilibrio solo mide alrededor de cero y, por lo tanto, puede ser muy sensible y no necesita estar calibrado. Por contra, tienen en principio menor respuesta dinámica y, si bien se pueden automatizar mediante un servomecanismo, no se logra normalmente una respuesta rápida como en los de deflexión.

Según el tipo de relación entrada - salida, los sensores pueden ser de orden cero, de primer orden, de segundo orden o de orden superior.

El orden está relacionado con el número de elementos almacenadores de energía independientes que incluye el sensor, y repercute en su exactitud y velocidad de respuesta.

En la tabla 1.1 se recogen estos criterios de clasificación y se dan ejemplos de sensores de cada clase. Cualquiera de estas clasificaciones es exhaustiva y cada una tiene un interés particular para diferentes situaciones de medida. Ahora bien, para el estudio de un gran número de sensores se suele acudir a su clasificación de acuerdo con la magnitud medida. Se habla en consecuencia, de sensores de temperatura, presión caudal, humedad, posición, velocidad, aceleración, fuerza, par, etc. Sin embargo, esta clasificación difícilmente puede ser exhaustiva ya que la cantidad de magnitudes que se pueden medir es prácticamente inagotable. Piénsese, por ejemplo, en la variedad de contaminantes químicos en el aire o en el agua, o en la cantidad de proteínas diferentes que hay en el cuerpo humano y que interesa detectar.

| Criterio | Clases | Ejemplos |
|-------------------|----------------|---------------------------|
| Aporte de energía | Moduladores | Termistor |
| | Generadores | Termopar |
| Señal de salida | Analógicos | Potenciómetro |
| | Digitales | Codificador de posición |
| Modo de operación | De deflexión | Acelerómetro de deflexión |
| | De comparación | Servoacelerómetro. |

Tabla 1.1
Criterios de clasificación de Sensores.

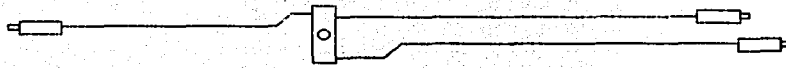


Desde el punto de vista de la ingeniería electrónica, la clasificación de los sensores es más atractiva de acuerdo con el parámetro variable: resistencia, capacitancia, inductancia, añadiendo luego los sensores generadores de tensión, carga o corriente, y otros tipos no incluidos en los anteriores grupos.

Si bien este tipo de clasificación es poco frecuente, es el elegido en este texto, pues permite reducir el número de grupos a unos pocos y se presta bien al estudio de los acondicionadores de señal asociados. En la tabla 1.2 se muestran los sensores y métodos de detección ordinarios para las magnitudes más frecuentes.

| Magnitud | Resistivos | Capacitivos | Inductivos y electro magnéticos | Generadores | Digitales | Uniones p-n | Ultrasonidos |
|--------------------------------------|---|----------------------------------|---|--------------------------------|---|------------------------------------|---|
| Posición distancia Desplazamiento | Potenciómetros Galgas magneto-resistencias | Condensador diferencial | LVDT Corriente Foucault Efecto Hall | | Codificadores incrementales y absolutos | Fotoeléctricos | Reflexión |
| Velocidad | | | Ley Faraday LVDT Efecto Hall corrientes Foucault | | Codificadores incrementales | | Efecto Doppler |
| Aceleración Vibración | Galgas + masa-resorte | | LVDT + masa-resorte | Piezoeléctricos + masa-resorte | | | |
| Temperatura | RTD Termistores | | | Termopares piroeléctricos | Osciladores de cuarzo | Diodo transistor convertidores T/I | |
| Presión | Potenciómetros + tubo Bourdon | Condensador variable + diafragma | LVDT + diafragma reluctancia variable + diafragma | Piezoeléctricos | Codificador + tubo Bourdon | | |
| Caudal flujo | Anemómetros de hilo caliente Galgas + voladizo termistores | | LVDT + rotámetro ley Faraday | | Vórtices | | Efecto Doppler tiempo tránsito vórtices |
| Nivel | Potenciómetro + flotador termistores LDR | Condensador variable | LVDT + flotador corrientes Foucault | | | Fotoeléctricos | Reflexión absorción |
| Fuerza | Galgas | Galgas capacitivas | Magneto-elástico LVDT + célula carga | Piezoeléctricos | | | |
| Humedad | Humistor | Dielectrico o variable | | | SAW | | |

Tabla No. 1.2
Sensores y métodos de detección.



1.4. - Características de los sensores.

1.4.1. - Exactitud, fidelidad, sensibilidad.

La exactitud "accuracy" en inglés es la cualidad que caracteriza la capacidad de un instrumento para dar indicaciones que se aproximen al verdadero valor de la magnitud medida.

La exactitud de un sensor se determina mediante la denominada calibración estática.

Esta consiste en mantener todas las entradas excepto una a un valor constante. La entrada en estudio se varía entonces lentamente, tomando sucesivamente valores "constantes" dentro del margen de medida, y se van anotando los valores que toma la salida.

La representación de estos valores en función de los de la entrada define la curva de calibración. Para poder conocer el valor de la magnitud de entrada esta debe tener un valor bien conocido, constituyendo lo que se denomina un patrón de referencia.

Su valor debe conocerse con una exactitud al menos diez veces mayor que la del sensor que se calibra.

La discrepancia entre la indicación del instrumento y el verdadero valor de la magnitud medida se denomina - error -. La diferencia entre la indicación del instrumento y el verdadero valor se denomina - error absoluto -.

A veces se da como porcentaje respecto al valor máximo que puede medir el instrumento o con respecto a la diferencia entre el valor máximo y el valor mínimo mensurables.

La fidelidad también llamada repetibilidad o precisión es la cualidad que caracteriza la capacidad de un instrumento de medida de dar el mismo valor de la magnitud medida, al medir varias veces en unas mismas condiciones determinadas (ambientales, operador, etc.), prescindiendo de su concordancia o discrepancia con el valor real de dicha magnitud.

La fidelidad implica que se tenga simultáneamente una conformidad en las lecturas sucesivas y un número alto de cifras significativas y es, por tanto, una condición necesaria y suficiente para la exactitud.



La sensibilidad o factor de escala es la pendiente de la curva de calibración, que puede ser o no constante a lo largo de la escala de medida. Para un sensor cuya salida este relacionada con la entrada x mediante la ecuación $y = f(x)$, la sensibilidad en el punto x_s , $S(x_s)$, es

$$S(x_s) = dy / dx$$

cuando $x = x_s$

En los sensores interesa tener una sensibilidad alta y si es posible, constante. Para un sensor con respuesta

$$y = kx + b$$

La sensibilidad es $S = k$, para todo el margen de valores de x aplicables. Para uno cuya respuesta sea

$$y = kx^2 + b$$

La sensibilidad es $S = 2kx$, y varía a lo largo de todo el margen de medida.

1.4.2. - Linealidad, resolución.

Las tres características anteriores - exactitud, fidelidad y sensibilidad - son suficientes para describir el comportamiento estático de un sensor. A veces se emplean en su lugar otras que expresan características alternativas o de interés particular para ciertos casos o bien son complementarias para conocer la idoneidad de un sistema de medida para una aplicación dada.

La Linealidad expresa el grado de coincidencia entre la curva de calibración y una línea recta determinada. Según cual sea la recta se habla de los siguientes casos:

- Linealidad independiente: la línea de referencia se define por el método de mínimos cuadrados. De esta forma, el máximo error positivo y el mínimo error negativo son iguales. Es la forma de especificación que suele dar mejores resultados.
- Linealidad ajustada al cero: la recta se define también por el método de los mínimos cuadrados, difiere del anterior, por la restricción adicional de pasar por cero.
- Linealidad terminal: la recta se define por la medición de la salida con entrada nula (o la menor del posible margen de medida) Y la salida teórica máxima, correspondiente a la mayor entrada admitida.
- Linealidad a través de los extremos: la recta se define mediante la salida real cuando la entrada al sensor es la menor del alcance especificado, y la salida real cuando la entrada es la máxima del alcance especificado.



- Linealidad teórica: la recta es la definida por las previsiones teóricas formuladas al diseñar el sensor.

En la figura 1.1 se representan estas distintas rectas para un sensor con una curva de calibración dada. Resulta que la Linealidad expresa hasta qué punto la sensibilidad es constante.

Para que un sensor sea útil no es necesario que sea lineal. El interés de la Linealidad esta en que la conversión lectura - valor medido es más fácil si la sensibilidad es constante, entonces basta multiplicar la indicación de salida por un factor constante para conocer el valor de entrada.

En instrumentos lineales la no Linealidad equivale a inexactitud.

Actualmente, con la posibilidad de incorporar un microprocesador en los sistemas de medida, interesa más la repetibilidad que la Linealidad, siempre es posible crear una tabla conteniendo los valores de entrada que correspondan a los valores de salida detectados.

Mediante una interpolación adecuada, es posible reducir el tamaño de dicha tabla.

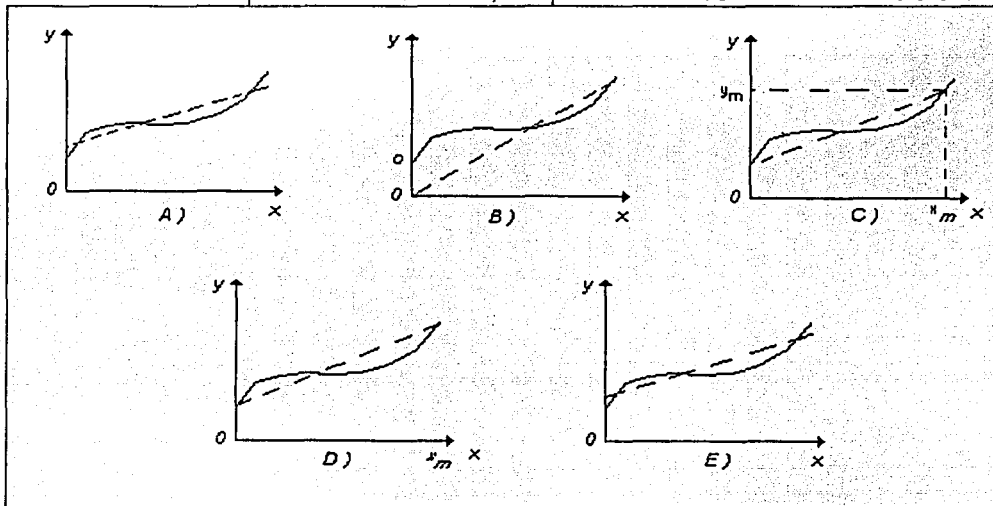


Figura 1.1

Rectas de referencia tomadas para definir la Linealidad.

A) mínimos cuadrados; B) mínimos cuadrados ajustada al cero; C) terminal; D) a través de los extremos; E) teórica.

Los principales factores que influyen en la Linealidad son: la resolución, el umbral y la histéresis.

Resolución.- Mínimo valor que se puede determinar en una escala.

Umbral.- Valor extremo o valor mínimo de una magnitud de la que depende un fenómeno, a partir del cual no ocurre o no es perceptible dicho fenómeno.



Sensibilidad.- Es el incremento mínimo de la entrada para el que se obtiene un cambio en la salida.

La histéresis se refiere a la diferencia en la salida para una misma entrada, según la dirección que tome la medición. Es decir, que puede suceder análogamente a la magnetización de los materiales ferromagnéticos, donde la salida correspondiente a una entrada depende de sí la entrada previa fue mayor o menor que la entrada actual.

Antes de pasar a la sección 1.5 donde se documenta la configuración de los sensores implementados, cabe aclarar que los sensores fotoeléctricos que utilizaremos, no se ven afectados por sus características dinámicas, dado que se calibraran las lecturas vía software, de manera tal, que podamos aproximar los dígitos con la precisión que necesite la medición.

1.5. - Forma de conexión de los sensores implementados.

Los sensores desarrollados para este trabajo fueron tres y todos fueron desarrollados con amplificadores operacionales, con el efecto de conseguir un diseño pequeño y fácil de manipular, además de confiable.

Los sensores tienen las siguientes etapas:

- a) Emisión y Recepción de Infrarrojo.- en esta etapa el sensor envía una fuente constante de radiación infrarroja con el dispositivo KR 8803, este a su vez tiene un fototransistor integrado en el mismo encapsulado. El fototransistor es capaz de recibir su propia señal a través de la terminal de la base que es sensible a la luz infrarroja.
- b) Buffer de Señal.- Para esta parte del proceso se toma la señal del sensor y simplemente se retransmite a la siguiente etapa, esto es con el objeto de proteger al fototransistor y evitar que aporte mas corriente de colector de lo que le permite su diseño de fábrica.
- c) Comparador no Inversor.- Esta etapa es la más importante del proceso. Es aquí donde con una configuración de comparador, se detecta si hay o no una variación de voltaje en la base del fototransistor. Cuando el voltaje cambia de 0 a algunos milivolts, el comparador se dispara y envía un valor de saturación, que es cercano a los 5 Volts y parece ser un pulso de directa.
- d) Preparación de Señal.- En esta etapa se acondiciona el valor de voltaje obtenido de la etapa anterior. Primeramente el pulso se suaviza con otro buffer y finalmente con un transistor en configuración de Emisor Común se invierte la señal de manera que el voltaje final del sensor es el adecuado para que el micro



lo interprete, además el transistor se encarga de reforzar la corriente de salida que nos da el amplificador operacional, ya que dichos dispositivos generalmente no proporcionan mucha corriente.

- e) Imagen Visual.- A la par que se acondiciona la señal para ser enviada al Buffer de tres estados, también se acondiciona para que manualmente con el Led de color verde se puedan calibrar los Sensores.

Estos Leds verdes, se encuentran en la parte posterior de los sensores y se encuentran encendidos, cuando no haya una pieza para medir en su campo - visual-. Estos Leds se apagan en el momento en que una pieza cruza el haz infrarrojo que emiten y se encienden al terminar de pasar la pieza.

A continuación en la Figura 1.2 se presenta el diagrama típico de un sensor, para observar las etapas descritas anteriormente y todos sus circuitos.

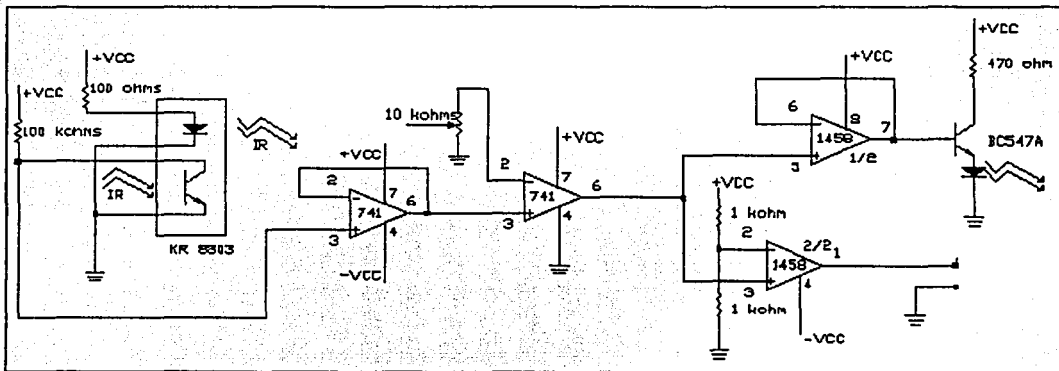
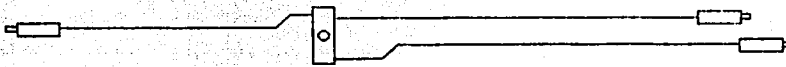


DIAGRAMA DE LOS SENSORES OPTICOS

Figura 1.2

El voltaje de polarización +Vcc es de 5v y -Vcc es de -5 Volts. De esta manera es más sencillo lograr el intercambio de voltajes con el microcontrolador.

**TESIS CON
FALLA DE ORIGEN**



2. - LENGUAJES DE ALTO NIVEL.

En este capítulo se analizan los lenguajes de programación que se pueden utilizar en la aplicación de un sistema de software. El diseño de lenguajes es de vital importancia para un proyecto de ingeniería que incluya manejo de software. Los lenguajes de programación son así instrumentos básicos.

Es importante tener una retroalimentación sobre la confiabilidad y utilidad de los lenguajes de programación en la práctica.

Un ingeniero con manejo de software debe tener cierto conocimiento de los lenguajes de programación para poder decidir sobre el lenguaje más adecuado a sus aplicaciones particulares.

Aquí se describirán distintas clases de lenguajes de programación y después se analizarán algunas de sus características principales, con el fin de optar por el lenguaje más adecuado.

2.1. - Clasificación de los lenguajes de programación.

Desde 1960, se han diseñado y aplicado miles de lenguajes de programación. La mayoría de estos lenguajes se han aplicado formando parte de proyectos de investigación y solo un número relativamente pequeño de ellos ha alcanzado un uso generalizado.

Pascal y C se usan en la programación de sistemas.

También existen aplicaciones científicas basadas en FORTRAN y aplicaciones profesionales en COBOL.

Además existen otros lenguajes como LISP y PROLOG que se utilizan en aplicaciones de inteligencia artificial.

En general, los lenguajes de programación se incluyen en una de las cinco categorías siguientes:

1. - *Lenguajes ensambladores.* Son lenguajes de máquina con una correspondencia uno a uno entre las proposiciones del lenguaje de programación y las operaciones de la máquina. La programación en código ensamblador requiere que el programador traduzca el diseño de su programa a secuencias de acciones de la máquina. Como estas notaciones son tan distintas, la programación en lenguaje ensamblador tiende a ser difícil y propensa a errores. Los lenguajes ensambladores son específicos de la familia de computadoras en los que se aplica.

2. - *Lenguajes para aplicación de sistema.* Esta clase de lenguajes de programación evoluciona a partir de los lenguajes ensambladores cuando se reconoció la dificultad de programar en ese código. Los lenguajes para aplicación de sistemas proporcionan algunas ventajas como las proposiciones de control y revisión de los tipos de variables, también permiten al operador tener acceso directo a las operaciones y direcciones de la máquina. Por lo tanto, en teoría todo lo que se



pueda programar en lenguaje ensamblador también se puede programar en un lenguaje para aplicación de sistemas.

3. - *lenguajes estáticos de alto nivel.* Son lenguajes que brindan al programador algunas proposiciones de control y declaración de variables, sin embargo, no brindan la posibilidad de controlar las operaciones de la máquina generadas por el compilador. Se caracterizan por tener asignación estática de la memoria. El espacio de memoria requerido para las variables del programa puede ser calculado por el compilador del lenguaje y reservado con anterioridad a la ejecución del programa. Aunque esto tiene algunas ventajas de aplicación para el escritor del lenguaje, impone restricciones al programador. Los lenguajes de alto nivel de esta clase fueron de los primeros, por lo que tuvieron amplia utilización. Los ejemplos más conocidos de estos tipos de lenguajes son COBOL y FORTRAN.

4. - *Lenguajes de alto nivel estructurados por bloques.* Estos lenguajes se desarrollaron a partir de los lenguajes estáticos, y proporcionan al programador una serie de construcciones de control y la posibilidad de clasificar los objetos de los programas dentro de un tipo determinado. Se distinguen por la particularidad de ofrecer una forma limitada de asignación dinámica de memoria llamada estructura de bloque. El compilador del lenguaje no puede tomar todas las decisiones sobre la cantidad de memoria que requieren las variables de un programa, y la ejecución de este se apoya en un sistema de administración de memoria, que puede asignar una cantidad de memoria a la entrada y retirarla a la salida de un bloque del programa.

Un bloque de programa es un área completamente delimitada y la ejecución del programa se interrumpe al entrar a un bloque para asignar la memoria. El programador debe saber con exactitud cuanta memoria se requiere al entrar en un bloque. Hasta no hace mucho, los lenguajes de esta clase no eran muy utilizados en aplicaciones comerciales, esta situación está cambiando a medida que se reconocen las ventajas que los lenguajes estructurados por bloques tienen sobre COBOL y FORTRAN. ALGOL y Pascal son ejemplos de esta clase de lenguajes.

5. - *Lenguajes dinámicos de alto nivel.* Esta clase de lenguajes de programación se distingue por el requisito siguiente, todo el control de la memoria se hace dinámicamente, esto es, la ejecución de las proposiciones individuales del lenguaje puede hacer que se asigne y retire la memoria. En general, la estructura de estos lenguajes tiende a ser bastante distinto de la estructura de los lenguajes de alto nivel estáticos o estructurados por bloques. En realidad, los lenguajes de esta clase rara vez se parecen. Los lenguajes dinámicos tienden a diseñarse para una aplicación particular y no son lenguajes de propósito general.

En general, hay una correspondencia de uno a muchos entre los lenguajes de programación de alto nivel y el código de máquina. Esto significa que se generan varias instrucciones de máquina para cada instrucción del lenguaje de alto nivel. Además de esto, los lenguajes de alto nivel permiten al usuario asociar nombres significativos a las variables del programa y a las subrutinas donde los nombres de



los objetos se relacionan con la entidad representada por ese objeto. En consecuencia, los programas de alto nivel son más fáciles de leer, escribir, depurar y mantener que los programas en código ensamblador.

2.2. - Construcciones para el manejo de estructuras de control.

El principal debate sobre la programación estructurada, que estuvo en boga a finales de la década de 1960 y principios de la de 1970, giró alrededor de la proposición *goto* y de si se debía usar o no en los programas. Gran parte de este debate fue estéril, y en la actualidad está generalmente aceptado que las construcciones de control expresivas, como las expresiones *while*, *case*, etc. Deben ser de uso normal para expresar el flujo de control de un programa. Sin embargo, hay situaciones en donde el empleo de proposiciones *goto* mejora la legibilidad de un programa. Por lo que es innecesaria su prohibición indiscriminada. En esta sección se describirá el diseño de ciertas abstracciones de control para algunos lenguajes de programación. Se cubren en los siguientes temas, las construcciones para el control de ciclos, las construcciones de decisión y el manejo de excepciones.

2.2.1. - Construcciones para el control de ciclos.

Las construcciones para el control de ciclos son aquellas construcciones de control del lenguaje que especifican una proposición, o grupo de ellas, que en un programa se deben ejecutar varias veces.

El número de veces se puede especificar de manera explícita o puede depender de que una condición se haga verdadera. Las construcciones de ciclos más comunes son las proposiciones *for*, *while-do* y *repeat-until*.

La proposición *for* numérica se usa cuando se quiere ejecutar un ciclo un número dado de veces. Tiene la forma general:

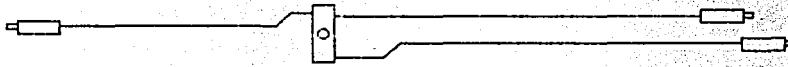
for {contador}: = { valor inicial} *step* {incremento} *to* {valor final}

Los distintos lenguajes difieren ligeramente en la naturaleza exacta de su proposición *for*.

La proposición *for* suele ser una construcción muy segura, porque en la mayoría de los lenguajes se garantiza la terminación del ciclo.

Las proposiciones *repeat* y *while* permiten al usuario especificar que el ciclo se debe ejecutar hasta que alguna condición sea verdadera o mientras la condición exista.

While {condición} *do* {proposición}
repeat {proposición} *until* {condición}



Estos tipos de ciclo no tienen los mismos problemas que algunos diseños de la proposición *for*, porque la variable de control del ciclo está bajo el control directo del programador, más que del sistema del lenguaje.

La proposición *while* permite al programador poner la prueba de terminación del ciclo al principio de este y la proposición *repeat* permite hacerlo al final.

Ninguna de las dos construcciones permite poner la prueba de terminación en la mitad del ciclo.

Hay muchas situaciones de programación prácticas donde es necesario poner la prueba de terminación dentro del ciclo.

La aplicación de la terminación desde dentro de un ciclo mediante el uso de proposiciones *if* y variables booleanas es pesada e incrementa la longitud del programa.

2.2.2. - Construcciones de decisión.

Las construcciones de decisión de un lenguaje de alto nivel son construcciones de control que permiten seleccionar una proposición o un grupo de ellas para su ejecución partiendo de la base de que alguna condición sea verdadera. Estas construcciones abarcan el condicional unilateral (*if-then*), el condicional bilateral (*if-then-else*) y el condicional multilateral (*case*).

La forma de los condicionales unilateral y bilateral es familiar :

```

if { condición } then { proposición }
if { condición } then { proposición } else { proposición }

```

la utilización de estas proposiciones no tiene problemas prácticos.

La proposición *case* en uso tiene diversas formas, a diferencia de los condicionales unilateral y bilateral, cada forma tiene algunos inconvenientes que pueden ser la causa de ciertos tipos de errores de programación.

2.2.3.- Control de excepciones en un programa.

Se llama excepción a un error de cierto tipo o a un suceso inesperado que se produce durante la ejecución de un programa. Las excepciones pueden ser originadas por errores de *hardware* o *software* que el programador puede o no haber detectado.

En general, cuando el programador no ha anticipado de manera explícita una excepción, el control se transfiere a un mecanismo controlador de excepciones del sistema que manipula la excepción. Si esta es seria se termina el programa en ejecución. Si el programador anticipó una excepción, debe incluir un código para detectarla y para tomar las medidas apropiadas cuando surja esa excepción. En general, si un programa se ejecuta bajo un sistema operativo, el programador no podrá detectar o manipular las excepciones del *hardware* de su programa; es misión del sistema operativo hacerse cargo de esos casos.



2.3.- Ventajas del Lenguaje Visual Basic 3.0 Profesional Para Windows.

La ventaja principal del Lenguaje Visual Basic así como de otros lenguajes de programación visuales que operan bajo el sistema operativo de Windows es el manejo amigable que le permite al usuario utilizar más fácilmente un programa. En este caso hizo posible la programación del puerto serial de la computadora de manera tal que el usuario lo estará utilizando para comunicarse con el microcontrolador de la banda transportadora con el único requerimiento de indicar en que número de puerto conectara el dispositivo.

Este lenguaje tiene recursos muy prácticos para el manejo de la información y de los comandos típicos del sistema operativo Windows. El programa ejecutable tiene la ventaja de poder operar correctamente en versiones de Windows superiores a la 3.1, que fue la versión en la cual se desarrolló.

Por estas razones se consideró adecuado utilizar este lenguaje de programación para desarrollar el software de la computadora personal.

2.4. - Programa manejador de la banda transportadora.

El *software* que maneja el prototipo de la banda transportadora se divide en dos programas, Estos son: el *software* de la computadora personal que es manipulado por el usuario y por otra parte el *software* del microcontrolador.

En el capítulo 3 se describe el programa del microcontrolador, ahora se indicarán los alcances del *software* de la computadora personal, que son los siguientes puntos:

- La pantalla de inicio informa al usuario que esta corriendo el programa.
- El programa recibe la información del usuario sobre el puerto en que se encuentra conectado el transportador.
- Da al usuario la opción de calibrar por software la precisión de las mediciones.
- En la pantalla de la computadora Personal se despliegan las dimensiones X, Y y Z de la pieza medida, en tiempo real.
- Las mediciones tomadas en cada sesión son guardadas en un archivo con formato para manejo de datos el cual es compatible con Excel de cualquier versión y puede ser abierto por el programa QUALITY CONTROL 4/e SQC 1.0 Student Versión, incluido en el libro Control De Calidad de Dale H. Besterfield mencionado en esta bibliografía. Para utilizar el archivo con este programa se debe escoger con qué dimensión (X, Y o Z) se efectuará el análisis estadístico, para borrar las otras dos dimensiones del archivo.
- El usuario puede detener el programa en cualquier momento que lo considere adecuado. Antes de iniciar la medición tiene la opción de regresar a cualquier



configuración del inicio del programa para corregir alguna opción si así lo desea.

EL programa de la computadora personal se encuentra en el archivo ejecutable llamado "BAND1.EXE" y se divide en tres ventanas o pantallas, donde el usuario puede recibir la información del prototipo y a su vez enviar información sobre el inicio de mediciones y el paro de las mismas. Las ventanas de las que se compone el programa son las siguientes:

a) VENTANA DE ENTRADA O DE INICIO.-

Esta pantalla es para mostrar al usuario que el programa inicia su ejecución y da algunos datos generales.

Existen en esta ventana dos botones con las siguientes opciones:

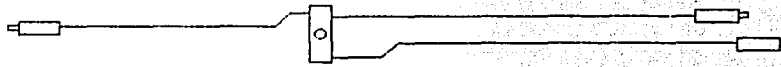
- CONTINUA.- con este botón el usuario puede continuar a la siguiente ventana.
- SALIR.- con este botón el usuario puede terminar la ejecución del programa en el momento que lo considere conveniente.



Figura 2.1

En la Figura 2.1. se observa la ventana de entrada o inicio.

TESIS CON
FALLA DE ORIGEN



VENTANA DE SELECCIÓN DE PUERTOS.-

Existen en esta ventana siete botones con las siguientes opciones:

- COM 1.- con este botón el usuario puede escoger el puerto serial 1 para comunicar al prototipo.
- COM 2.- con este botón el usuario puede escoger el puerto serial 2 para comunicar al prototipo.
- COM 3.- con este botón el usuario puede escoger el puerto serial 3 para comunicar al prototipo.
- COM 4.- con este botón el usuario puede escoger el puerto serial 4 para comunicar al prototipo.
- CONTINUA.- con este botón el usuario puede continuar a la siguiente ventana.
- REGRESA.- con este botón el usuario puede regresar a la ventana previa, por si requiere cambiar alguna configuración.
- SALIR.- con este botón el usuario puede terminar la ejecución del programa en el momento que lo considere conveniente.

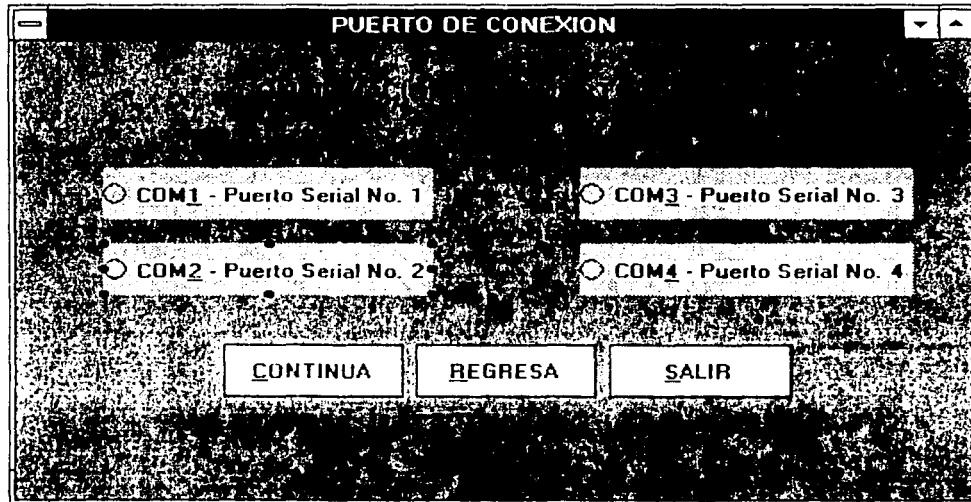


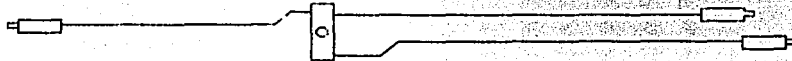
Figura 2.2

En la Figura 2.2. se observa la ventana de selección de puerto.

c) VENTANA DE MEDICIÓN.-

Existen en esta ventana cuatro botones con las siguientes opciones:

- CALIBRAR.- con este botón el usuario puede calibrar la resolución de la medida que reciban los sensores.
- INICIAR, CONTINUAR MEDICION.- con este botón el usuario puede iniciar o continuar el proceso de medición.



- REGRESA.- con este botón el usuario puede regresar a la ventana previa, por si requiere cambiar alguna configuración.
- SALIR.- con este botón el usuario puede terminar la ejecución del programa en el momento que lo considere conveniente.

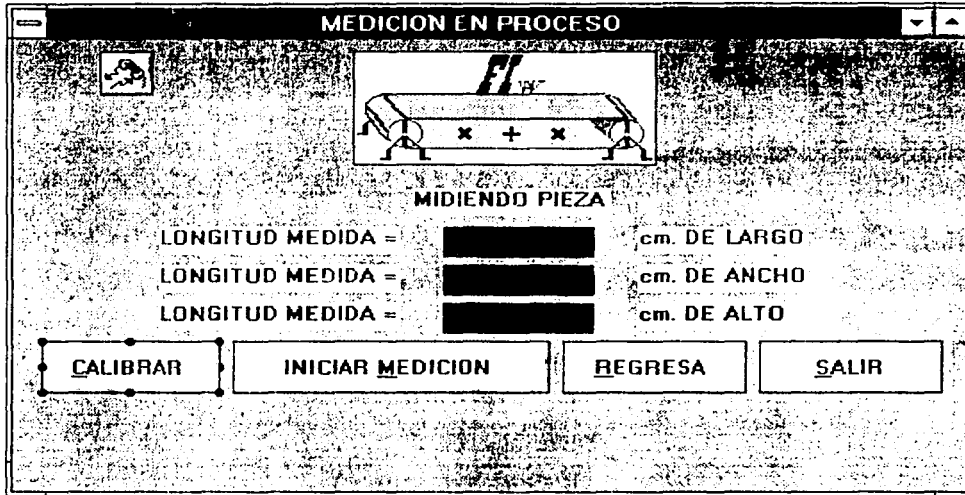


Figura 2.3

En la Figura 2.3. se observa la ventana de medición.

A continuación se presenta el código fuente del programa que maneja el intercambio de puertos seriales:

```

Sub CONTINUA_Click ()
  If (SERIAL1 Or SERIAL2 Or SERIAL3 Or SERIAL4) = 0 Then
    For i = 1 To 10 ' RUTINA PARA EVITAR QUE USUARIO
      Beep ' PASE SIN SELECCIONAR UN PUERTO
    Next i
    MsgBox "DEBE SELECCIONAR PUERTO", 16, "AVISO"
  Else
    PUERTO.Hide ' ESCONDE PUERTO.FRM
    ' FORMEDICION.Show ' MUESTRA FORMEDIC.FRM EN REVISION PARA BORRAR
    MEDICION.Show ' MUESTRA VENTANA DE MEDICIONES
  End If
  ' INICIALIZACION DE ARCHIVO PARA BASE DE DATOS
  ARCHIVOLIBRE% = FreeFile ' TOMA NÚMERO DISPONIBLE DEL SISTEMA
  Open "MEDICION.DAT" For Output As #ARCHIVOLIBRE% ' CONFIGURA ARCHIVO
  Print #ARCHIVOLIBRE%, " X Y Z" ' ESCRIBE EN ARCHIVO
  Close #ARCHIVOLIBRE% ' CIERRA ARCHIVO
  ' INICIALIZA EL NÚMERO DE PIEZAS DEL CONTADOR

  numerodepieza = 1 ' ESTE CONTADOR APARECERA EN EL TEXTO DE MEDICION
End Sub

```

TESIS CON
FALLA DE ORIGEN



Así mismo, a continuación se presenta el código fuente del programa que ejecuta e interpreta las mediciones de la banda transportadora.

Sub CONTINUA_Click ()

```

'=====
'  RUTINA PRINCIPAL PARA MOSTRAR MEDICIONES EN PANTALLA
'=====

REGRESA.Enabled = False ' NO SE PERMITE REGRESAR
CALIBRAR.Enabled = False ' YA NO SE PUEDE CALIBRAR
CONTINUA.Caption = "CONTINUAR &MEDICION" ' CAMBIA
                    ' LEYENDA EN EL BOTON DE LA VENTANA
CONTINUA.Enabled = False' DESHABILITA EL BOTON CONTINUA

'=====
' CONFIGURACIONES DEL PUERTO DE APERTURA
'=====

comm1.Settings = "2400,n,8,1" ' CONFIGURACIONES PUERTO
comm1.InputLen = 0 ' LEE TODO EL BUFFER DE ENTRADA
comm1.PortOpen = True ' ABRE EL PUERTO

'=====
' INICIO DE TRANSMISION
'=====

comm1.Output = "H" ' SE MANDO UNA -H-

'=====
' LECTURA DE BUFFER DE ENTRADA
'=====

Do
    split = DoEvents() ' SE ESPERA RESPUESTA EN EL PUERTO
    Loop Until comm1.InBufferCount >= 1
    nueva$ = comm1.Input ' DATOS SE GUARDAN EN -NUEVA$-

'=====
' INTERPRETACION DE BUFFER DE ENTRADA
'=====


If nueva$ = "HW" Then
    comm1.PortOpen = False' CIERRA EL PUERTO
    MsgBox ("COMUNICACION SERIAL Y SENSORES TRABAJANDO OK"), 64
    CONTINUA.Enabled = True ' HABILITA CONTINUA

ElseIf nueva$ = "HX" Then
    comm1.PortOpen = False' CIERRA EL PUERTO
    MsgBox ("SENSOR DE LONGITUD -X- DESALINEADO O DESCONECTADO,
    VERIFICAR Y REINICIAR MICRO Y EL PROGRAMA EN LA PC"), 16
    End ' TERMINA EL PROGRAMA

ElseIf nueva$ = "HY" Then
    comm1.PortOpen = False' CIERRA EL PUERTO
    MsgBox ("SENSOR DE ANCHO -Y- DESALINEADO O DESCONECTADO, VERIFICAR
    Y REINICIAR MICRO Y EL PROGRAMA EN LA PC"), 16
    End ' TERMINA EL PROGRAMA

ElseIf nueva$ = "HZ" Then
    comm1.PortOpen = False' CIERRA EL PUERTO

```



```
MsgBox ("SENSOR DE ALTURA -Z- DESALINEADO O DESCONECTADO,  
VERIFICAR Y REINICIAR MICRO Y EL PROGRAMA EN LA PC"), 16  
End ' TERMINA EL PROGRAMA
```

```
ElseIf (nueva$ < Chr$(48) Or nueva$ > Chr$(48)) Then  
comm1.PortOpen = False' CIERRA EL PUERTO  
MsgBox ("ERROR EN LA INTERFAZ DE COMUNICACION SERIAL, VERIFICAR Y  
REINICIAR MICRO Y EL PROGRAMA EN LA PC"), 16  
End ' TERMINA EL PROGRAMA
```

```
End If
```

```
'=====
```

```
' CONTADOR DEL NÚMERO DE PIEZAS EN MEDICION
```

```
'=====
```

```
Text1.Text = numerodepieza
```

```
'=====
```

```
' RECEPCION DE DATOS DE MEDICION
```

```
'=====
```

```
For i = 1 To 1000 ' CICLO ANIDADO PARA OBTENER UN RETARDO  
For j = 1 To 100: Next j ' EN LO QUE SE LLEVA A CABO LA  
Next i ' MEDICION EN LA BANDA TRANSPORTADORA
```

```
comm1.PortOpen = True ' ABRE EL PUERTO POR SEGUNDA OCASION  
Do
```

```
solomillo = DoEvents()  
Loop Until comm1.InBufferCount >= 7  
lectura$ = comm1.Input ' DATOS SE GUARDAN EN -lectura$-
```

```
comm1.PortOpen = False' CIERRA EL PUERTO
```

```
'=====
```

```
' VERIFICACION DE LA LECTURA DEL BUFFER
```

```
'=====
```

```
If Mid$(lectura$, 1, 1) <> "@" Then ' COMPROBACION DE LA -@-  
MsgBox ("ERROR EN LA CODIFICACION DE LA INFORMACION, EFECTUE  
MEDICIONES NUEVAMENTE Y VERIFIQUE QUE EL SISTEMA ESTE NORMAL"), 16  
End ' TERMINA EL PROGRAMA
```

```
Else
```

```
previo_x$ = Mid$(lectura$, 2, 1) ' ASIGNACION VALOR PREVIO DE X  
desbor_x$ = Mid$(lectura$, 3, 1) ' ASIGNACION DEL VALOR DEL  
DESBORDAMIENTO X  
previo_y$ = Mid$(lectura$, 4, 1) ' ASIGNACION VALOR PREVIO DE Y  
desbor_y$ = Mid$(lectura$, 5, 1) ' ASIGNACION DEL VALOR DEL  
DESBORDAMIENTO Y  
previo_z$ = Mid$(lectura$, 6, 1) ' ASIGNACION VALOR PREVIO DE Z  
desbor_z$ = Mid$(lectura$, 7, 1) ' ASIGNACION DEL VALOR DEL  
DESBORDAMIENTO Z
```

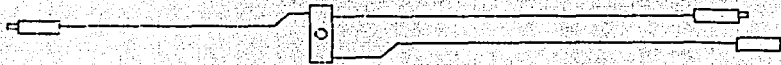
```
End If
```

```
'=====
```

```
' ASIGNACION DE LOS VALORES DE LAS MEDICIONES
```

```
'=====
```

```
medicion_x = Asc(previo_x$) - 33 ' CONVERSION A NÚMERO DEL CODIGO
```



```

' RECIBIDO
itera_x = Asc(desbor_x$) - 32 ' CONVERSION A NÚMERO DE LAS
' ITERACIONES
total_x = (222 * itera_x) + medicion_x ' CONVERSION A UN NÚMERO
' TOTAL

medicion_y = Asc(previo_y$) - 33 ' CONVERSION A NÚMERO DEL CODIGO
' RECIBIDO
itera_y = Asc(desbor_y$) - 32 ' CONVERSION A NÚMERO DE LAS
' ITERACIONES
total_y = (222 * itera_y) + medicion_y ' CONVERSION A UN NÚMERO
' TOTAL

medicion_z = Asc(previo_z$) - 33 ' CONVERSION A NÚMERO DEL CODIGO
' RECIBIDO
itera_z = Asc(desbor_z$) - 32 ' CONVERSION A NÚMERO DE LAS
' ITERACIONES
total_z = (222 * itera_z) + medicion_z ' CONVERSION A UN NÚMERO
' TOTAL

text3.Text = Str$(total_x)
text7.Text = Str$(total_y)
text8.Text = Str$(total_z)

' =====
' ESCRITURA DE VALORES DE MEDICION EN ARCHIVO
' =====
ARCHIVOLIBRE% = FreeFile ' TOMA NÚMERO DISPONIBLE DEL SISTEMA
Open "MEDICION.DAT" For Append As #ARCHIVOLIBRE% ' CONFIGURA EL
' ARCHIVO PARA ESCRITURA Y LO NOMBRA
Print #ARCHIVOLIBRE%, ; total_x; ; total_y; ; total_z: ' ESCRIBE EL
' NOMBRE DE LAS COORDENADAS EN EL ARCHIVO
Close #ARCHIVOLIBRE% ' TEMPORAL MIENTRAS SE DEPURA LA RUTINA

CONTINUA.Enabled = True ' habilita CONTINUA

numerodepieza = numerodepieza + 1 ' INCREMENTA EL NÚMERO DE PIEZAS
' MEDIDAS EN LA PANTALLA
End Sub

```

Estos códigos son medulares en la ejecución del programa para comunicarse con el microcontrolador 87C51 y manejar el encendido y apagado del motor.



3.- MICROCONTROLADORES Y CIRCUITOS DIGITALES.

Como preámbulo al tratamiento de los circuitos digitales se presentará la definición de lógica binaria de manera tal que exista una base sobre la cual los conceptos precedentes se puedan asentar.

3.1. - Definición de la lógica binaria.

La lógica binaria consta de dos elementos primordiales: variables binarias y operaciones lógicas.

Las variables se denotan con letras del alfabeto como A, B, C, x, y, z, etc.

Cada variable tiene dos y solo dos posibles valores distintos, 1 y 0.

Hay tres operaciones lógicas básicas: AND, OR y NOT.

1. - AND (\wedge): Esta operación se representa mediante un punto o por la ausencia de operador.

Por ejemplo:

$$x \wedge y = z \quad \text{ó} \quad xy = z \quad \text{se lee "x y es igual a z"}$$

La operación lógica AND se interpreta con el significado $z = 1$ si y solo si $x = 1$ y $y = 1$; en cualquier otro caso $z = 0$.

2. - OR (\vee): Esta operación se representa mediante un signo de suma.

Por ejemplo:

$$x + y = z \quad \text{se lee "x ó y es igual a z"}$$

Lo cual significa que $z = 1$ si $x = 1$ ó si $y = 1$ ó si tanto $x = 1$ como $y = 1$. Si tanto $x = 0$ como $y = 0$, entonces $z = 0$.

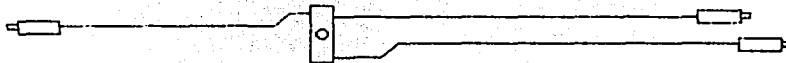
3. - NOT (NO): Esta operación esta representada por una sola comilla (y algunas veces por una barra encima de la variable).

Por ejemplo:

$$x' = z \quad \text{ó} \quad (x = \bar{z}) \quad \text{se lee "x no igual a z"}$$

significa que z es lo que x no es. En otras palabras si $x = 1$, entonces $z = 0$; y si $x = 0$, entonces $z = 1$.

La lógica binaria es semejante a la aritmética binaria y las operaciones AND y OR tienen ciertas similitudes con la multiplicación y la suma respectivamente. De hecho, los símbolos que se utilizan para AND y OR son los mismos que se usan para la multiplicación y la suma. Sin embargo, la lógica binaria no debe confundirse con la aritmética binaria. Debe tomarse en cuenta que una variable aritmética denota un número que puede constar de muchos dígitos. Una variable lógica siempre será un 1 ó 0.



3.2. - Circuitos digitales combinacionales.

Los circuitos lógicos para sistemas digitales pueden ser combinacionales o secuenciales. Un circuito consta de compuertas lógicas cuyas salidas en cualquier momento están determinadas en forma directa por la combinación presente de las entradas sin tomar en cuenta las entradas previas.

Un circuito combinacional realiza una operación específica de procesamiento de información, explicada por completo en forma lógica por un conjunto de funciones booleanas.

Los circuitos secuenciales emplean elementos de memoria (celdas binarias) además de las compuertas lógicas.

Un circuito combinacional consta de variables de entrada, compuertas lógicas y variables de salida. Las compuertas lógicas aceptan las señales de las entradas y generan señales a las salidas.

Este proceso transforma la información binaria de los datos dados de entrada en los datos requeridos de salida.

En la figura 3.1, se muestra un diagrama de bloques de un circuito. Las n variables binarias de entrada provienen de una fuente externa; las m variables de salida van también a un destino externo.

En muchas aplicaciones, la fuente y/o destino son registros de almacenamiento localizados ya sea en la proximidad del circuito o en un dispositivo externo remoto.

Para las n variables de entrada hay 2^n combinaciones posibles de los valores binarios de entrada.

Para cada combinación posible de entrada hay una y solo una posible combinación de salida.

Un circuito combinacional puede describirse por n funciones booleanas, una para cada variable de salida. Cada función de salida se expresa en términos de las n variables de entrada.



Figura 3.1

Diagrama de bloques de un circuito combinacional.

3.2.1. - Álgebra booleana.

En 1854 George Boole introdujo un tratamiento sistemático de la lógica y desarrolló para este propósito un sistema algebraico que ahora se conoce como álgebra booleana.



En 1938 C. E. Shannon introdujo un álgebra booleana de dos valores denominada álgebra de interruptores, en la cual demostró que las propiedades de los circuitos eléctricos y estables con interruptores, pueden representarse con esta álgebra.

El álgebra booleana es una estructura algebraica definida en un conjunto de elementos B junto con dos operadores binarios $+$ y \cdot siempre que satisfagan los postulados de Huntington (cierre en los operadores, elemento identidad, distribuidad y conmutatividad).

3.2.2. - Compuertas lógicas.

Ya que las funciones booleanas se expresan en términos de las operaciones AND, OR y NOT, es fácil implantar una función booleana con estos tipos de compuertas. La posibilidad de construir compuertas para otras operaciones lógicas es de interés práctico.

Dentro del álgebra booleana los operadores AND y OR son binarios, y NOT es unitario, esta clasificación se origina por las variables de entrada.

Si se combinan las 2 variables para las funciones binarias Tendremos $n = 2$ para 2^n de lo cual resulta un 4. Esto implica que existen cuatro diferentes combinaciones de valores que las variables pueden tomar simultáneamente.

Cuando consideramos que las funciones binarias son también 2, nuevamente ocurre que $n = 2$ para 2^n con este nuevo 4. Se entiende que las posibilidades de funciones que se pueden aplicar simultáneamente son cuatro también.

En esencia se tiene una potencia de potencias $2^{2^2} = 16$.

Por ello se pueden formar 16 combinaciones diferentes de funciones lógicas nuevas, formadas a partir de 2 variables y dos operadores binarios.

Para el operador unitario NOT nuestras combinaciones se reducen a $2^1 = 2$ por ello con este operador y dos variables lógicas, solo se obtienen dos combinaciones.


Sobre la base de lo anterior, se forman 18 funciones posibles. De estas funciones se utilizan las más prácticas para fines de diseño.

En la tabla 3.1 se presenta el nombre, tabla de verdad (relación de entradas y salida), función algebraica y símbolo gráfico, de las mismas así como los símbolos especiales de las compuertas lógicas Excluyente OR y Excluyente NOR.

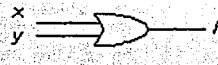


Tabla 3.1
Operaciones Lógicas, Símbolo de Compuertas y Tabla de verdad.


AND

| x | y | F |  |
|---|---|---|---|
| 0 | 0 | 0 | $F = xy$ |
| 0 | 1 | 0 | |
| 1 | 0 | 0 | |
| 1 | 1 | 1 | |

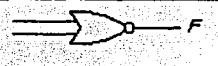
OR

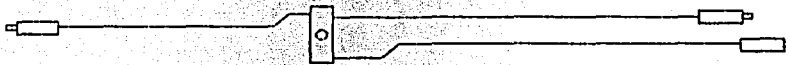
| x | y | F |  |
|---|---|---|---|
| 0 | 0 | 0 | $F = x+y$ |
| 0 | 1 | 1 | |
| 1 | 0 | 1 | |
| 1 | 1 | 1 | |

NAND

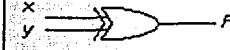
| x | y | F |  |
|---|---|---|---|
| 0 | 0 | 1 | $F = (xy)'$ |
| 0 | 1 | 1 | |
| 1 | 0 | 1 | |
| 1 | 1 | 0 | |

NOR


| x | y | F |  |
|---|---|---|---|
| 0 | 0 | 1 | $F = (x + y)'$ |
| 0 | 1 | 0 | |
| 1 | 0 | 0 | |
| 1 | 1 | 0 | |




XOR (Excluyente - OR)

| x | y | F |  |
|-----|-----|-----|---|
| 0 | 0 | 0 | $F = xy' + x'y$ |
| 0 | 1 | 1 | $F = x \oplus y$ |
| 1 | 0 | 1 | |
| 1 | 1 | 0 | |


NOR Excluyente o equivalente

| x | y | F |  |
|-----|-----|-----|---|
| 0 | 0 | 1 | $F = xy + x'y'$ |
| 0 | 1 | 0 | $F = x \odot y$ |
| 1 | 0 | 0 | |
| 1 | 1 | 1 | |

INVERSOR

| x | F |  |
|-----|-----|---|
| 0 | 1 | $F = x'$ |
| 1 | 0 | |

BUFFER

| x | F |  |
|-----|-----|--|
| 0 | 0 | $F = x$ |
| 1 | 1 | |

Como se puede observar, para fines prácticos, se utilizan solo 6 compuertas binarias y 1 unitaria con la función NOT.

Propiamente, el BUFFER no es una compuerta lógica, dado que no efectúa ninguna operación sobre su entrada. Su utilización es para revitalizar la señal cuando se maneja por cables relativamente largos o para conectar en las salidas de un microcontrolador, evitando así que este se force al entregar ciertos valores de voltaje.



3.3. - Lógica combinacional con MSI y LSI.

La lógica combinacional con compuertas es buena, solo que presenta algunos inconvenientes.

Dentro del mismo encapsulado o circuito integrado se encuentran generalmente cierto número de compuertas iguales.

Si un diseño requiere de diferentes tipos de compuertas se necesitarán prácticamente el mismo número de circuitos que de compuertas, desde luego hay que observar las condiciones particulares de cada diseño para que la generalización sea aplicable.

Una solución para obtener diseños más pequeños físicamente y a la vez económicos, se obtiene utilizando lógica combinacional con MSI y LSI.

MSI proviene de sus siglas en inglés (Medium Scale of Integration) ó mediana escala de integración, así mismo LSI (Large Scale of Integration) significa gran escala de integración.

Existe toda una serie de circuitos integrados de MSI y LSI en el mercado para facilitar la tarea de implementar diseños prácticos.

Estos circuitos internamente se constituyen de arreglos con diferentes compuertas, de manera tal que se puedan utilizar para diferentes diseños. A continuación mostraremos brevemente los principales circuitos que podemos encontrar en el mercado:

3.3.1. - Decodificadores.

Un código binario de n bits es capaz de representar hasta 2^n elementos distintos de información codificada.

Un decodificador es un circuito combinacional que convierte información binaria de n líneas de entrada a un máximo de 2^n líneas únicas de salida.

Si la información del decodificador de n -bit tiene combinaciones no usadas o que no importan, la salida del decodificador tendrá menos de 2^n salidas.

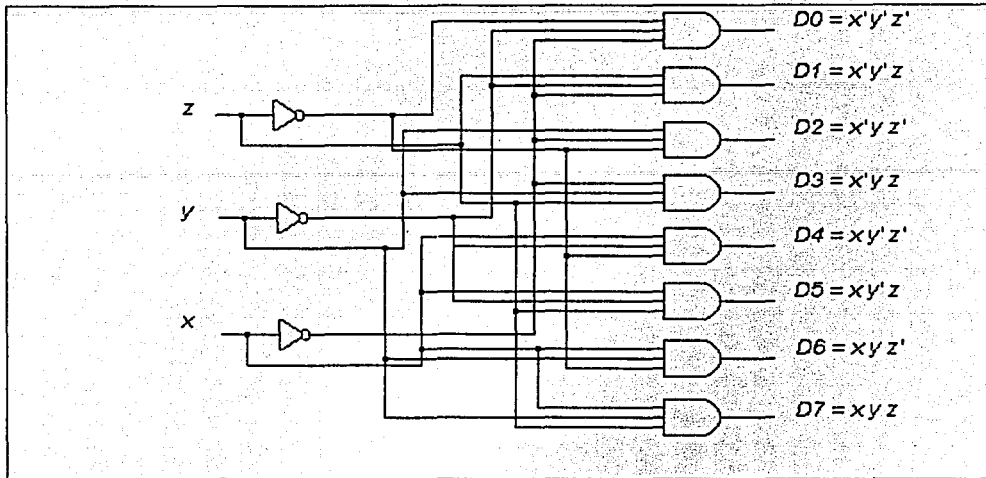


Figura 3.2.

La figura 3.2 muestra la constitución interna de un decodificador 3 a 8 líneas.

Para entender el funcionamiento lógico del diagrama anterior, aquí se presenta la tabla de verdad del decodificador.

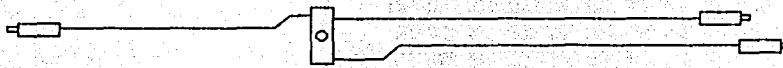
| Entradas | | | Salidas | | | | | | | |
|----------|---|---|---------|----|----|----|----|----|----|----|
| x | y | z | D0 | D1 | D2 | D3 | D4 | D5 | D6 | D7 |
| 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 1 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 1 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 |
| 0 | 1 | 1 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 |
| 1 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 |
| 1 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 |
| 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 |
| 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |

Tabla 3.2

Tabla de Verdad de un Decodificador

Como se puede observar en la tabla de verdad, a cada combinación de las entradas corresponde exclusivamente a una salida.

En la práctica la utilización de un multiplexor nos ahorra el uso de varias compuertas lógicas.



3.3.2. - Demultiplexores.

Ahora veremos un elemento similar al decodificador, con la variedad de operar bajo lógica invertida.

Esto significa que la salida de un demultiplexor, será un cero y no un uno lógico.

Algunos decodificadores en circuito integrado se construyen con compuertas NAND. Ya que esta compuerta produce la operación AND con una salida invertida, se vuelve más económico generar los términos del decodificador en su forma complementaria.

La mayoría si no todos los decodificadores incluyen una o más entradas de capacitación o habilitación (en inglés ENABLE = *E*) construida con compuertas NAND, se muestra en la figura 3.2 el diagrama lógico de un demultiplexor.

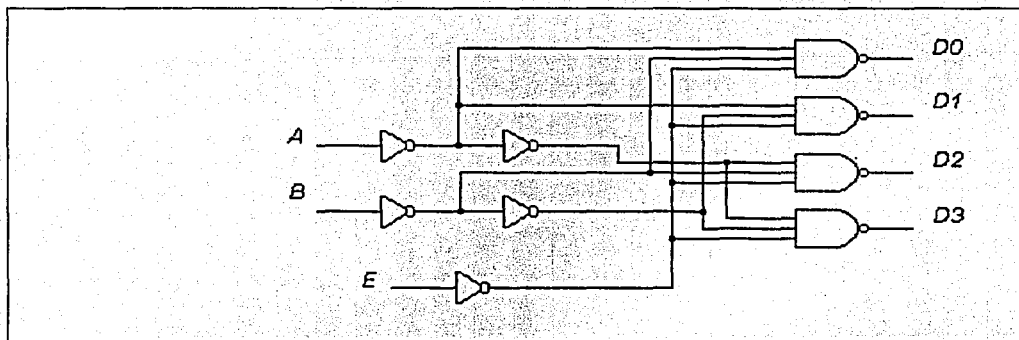


Figura 3.2

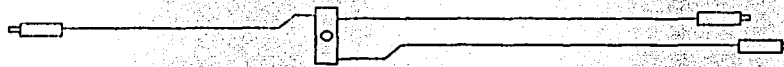
Diagrama de un Demultiplexor 2 a 4 Líneas con habilitación de Entrada(*E*)

La Figura 3.2 muestra un Demultiplexor de 2 a 4 Líneas y la Tabla 3.3 nos indica su operación lógica.

| <i>E</i> | <i>A</i> | <i>B</i> | <i>D0</i> | <i>D1</i> | <i>D2</i> | <i>D3</i> |
|----------|----------|----------|-----------|-----------|-----------|-----------|
| 1 | X | X | 1 | 1 | 1 | 1 |
| 0 | 0 | 0 | 0 | 1 | 1 | 1 |
| 0 | 0 | 1 | 1 | 0 | 1 | 1 |
| 0 | 1 | 0 | 1 | 1 | 0 | 1 |
| 0 | 1 | 1 | 1 | 1 | 1 | 0 |

Tabla 3.3

Operación Lógica de un Demultiplexor de 2 a 4 Líneas



La X de la tabla se llama condición NO IMPORTA (ò Don't care) y simboliza que para ese diseño o en esa parte, no importará si se tiene un cero o un uno, la salida no se alterará.

Esta condición es muy útil al momento de diseñar, su utilización será más adelante.

Se observa que cuando $E = 1$, no importa que valores tomen A ò B ya que el dispositivo se encuentra deshabilitado.

En general el comportamiento de un demultiplexor y el decodificador, es idéntico, según se observa, solo habrá que recordar bajo que lógica queremos que opere el dispositivo que se utilizará en el diseño.

3.3.3. - Codificadores.

Un codificador es una función digital que produce una operación inversa de la de un decodificador. Un codificador tiene 2^n (o menos) líneas de entrada y n líneas de salida.

Las líneas de salida generan el código binario para las 2^n variables de entrada.

Un ejemplo de decodificador se muestra en la figura 3.3.

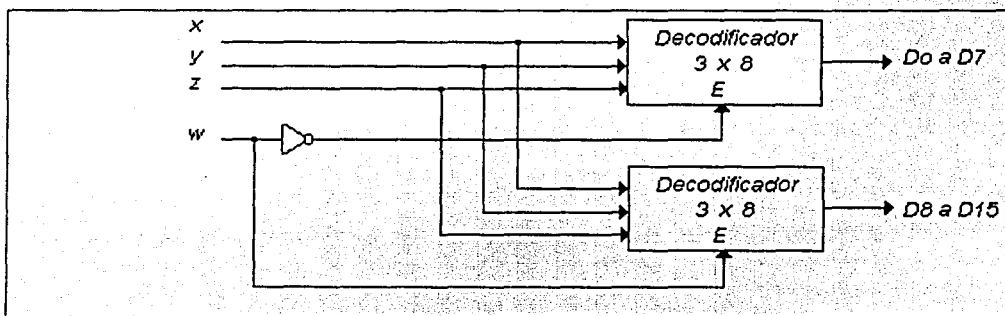


Figura 3.3

Diagrama de un Decodificador de 4 a 16 construido con dos Decodificadores 3 a 8 líneas y un inversor.

Los codificadores del tipo ilustrado no están disponibles en encapsulado de circuitos integrados, ya que pueden construirse fácilmente con compuertas OR. El tipo de codificador disponible en la forma de circuito integrado se denomina codificador de prioridad. Estos codificadores establecen una prioridad de entrada para asegurar que solo se codifique la línea de entrada de más alta prioridad.



| Entradas | | | | | | | | Salidas |
|----------|----|----|----|----|----|----|----|---------|
| D0 | D1 | D2 | D3 | D4 | D5 | D6 | D7 | x y z |
| 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 0 0 |
| 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 0 1 |
| 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 1 0 |
| 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 1 1 |
| 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 1 0 0 |
| 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 1 0 1 |
| 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 1 1 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 1 1 |

Tabla 3.4

Tabla de Verdad del Codificador de 4 a 16

Así que en la tabla 3.4, se da prioridad a una entrada con un número de subíndice más alto sobre uno con un número de subíndice más bajo.

3.3.4. - Multiplexores.

La multiplexión significa transmitir un gran número de unidades de información sobre un número más pequeño de canales o líneas. Un multiplexor digital es un circuito combinacional que selecciona información binaria sobre una de muchas líneas de entrada y la dirige a una sola línea de salida. La selección de una línea particular de entrada esta controlada por un conjunto de líneas de selección. En forma normal hay 2^n líneas de entrada y n líneas de selección de cuyas combinaciones se determinará cuál entrada se selecciona.

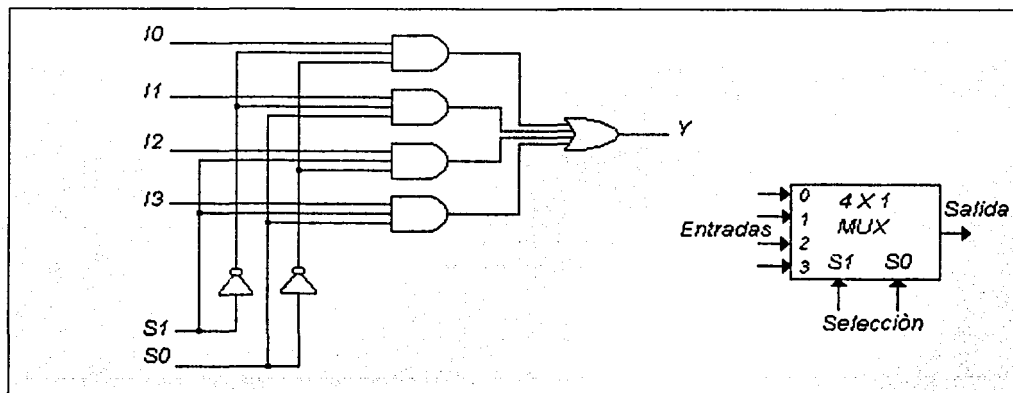
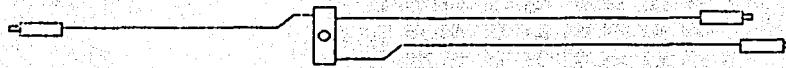


Figura 3.4

En la figura 3.4 se muestra un multiplexor de 4 líneas a 1 línea.



Cada una de las cuatro líneas 10, a 13, se aplica a una entrada de una compuerta AND. Las líneas de selección S1 y S0 se decodifican para seleccionar una compuerta AND particular.

| S1 | S0 | Y |
|----|----|----|
| 0 | 0 | 10 |
| 0 | 1 | 11 |
| 1 | 0 | 12 |
| 1 | 1 | 13 |

Tabla 3.5

La tabla 3.5 muestra la función de las líneas de selección en un multiplexor

3.3.5. - Memoria de solo lectura ROM.

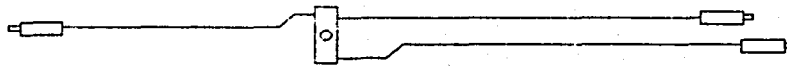
En la sección 3.3.1. se vió que un decodificador genera los 2^n términos de las n variables de entrada. Por la inserción de compuertas OR para la de los términos de las funciones Booleanas, se tuvo capacidad de generar cualquier circuito combinacional deseado.

Una memoria de solo lectura ROM (Read Only Memory) es un dispositivo que incluye tanto el decodificador como las compuertas OR dentro de un solo circuito integrado.

Las conexiones entre las salidas del decodificador y las entradas de las compuertas OR pueden especificarse para cada configuración particular por la "programación" de la ROM. Con mucha frecuencia se usa la ROM para implementar un circuito combinacional complejo en un paquete de circuito integrado y, en ese caso, elimina todos los alambres de interconexión.

Una ROM en forma esencial es un dispositivo de memoria (o almacén) en el cual se guarda un conjunto fijo de información binaria. La información binaria primero debe especificarla el usuario y entonces se inserta en la unidad para formar el patrón requerido de interconexión. Las ROM se obtienen con eslabones internos especiales que pueden fusionarse o romperse. La interconexión deseada para una aplicación particular requiere que se fusionen ciertos eslabones para formar las trayectorias del circuito necesario. Una vez que se establece un patrón para una ROM, permanece fijo aun cuando se enciende la potencia del circuito y se apague otra vez.

Las memorias de solo lectura constan de n líneas de entrada y m líneas de salida. Cada combinación de bits de las variables de entrada se denomina *dirección*. Cada combinación de bits que se obtiene de las líneas de salida, se conoce como *palabra*. El número de bits por palabra es igual al número de líneas de salida m . Una *dirección* es, en esencia, un número binario que denota uno de los términos de las n variables. El número de direcciones distintas posibles con n variables de entrada es 2^n .



Una palabra de salida puede seleccionarse por una dirección única, ya que hay 2^n direcciones distintas en una ROM. Hay 2^n palabras distintas que se dice se almacenan en la unidad. La palabra disponible en las líneas de salida en cualquier momento dado depende del valor de dirección aplicado a las líneas de entrada. Una ROM se caracteriza por el número de palabras 2^n y el número de bits por palabra m .

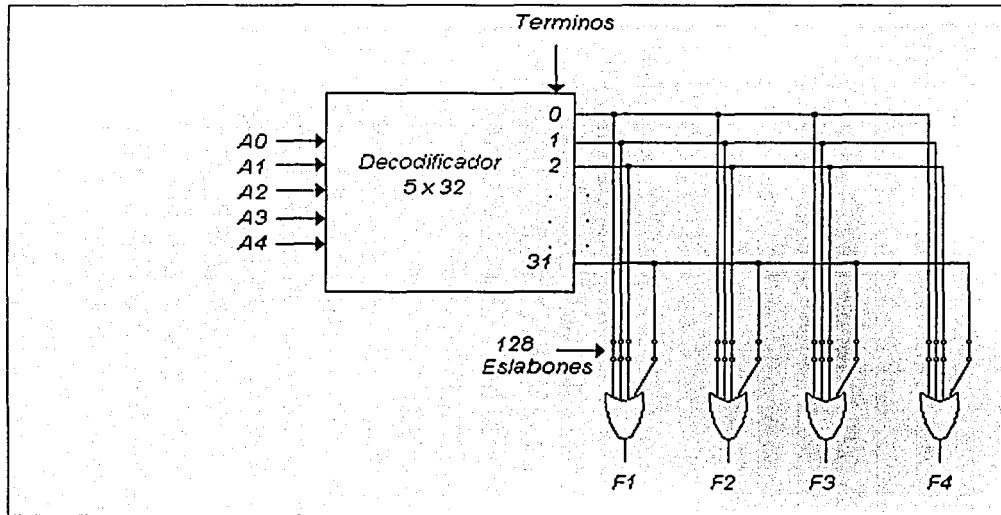


Figura 3.5

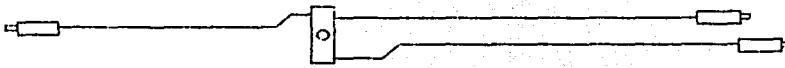
En la figura 3.5 se muestra la construcción lógica de una ROM de 32×4 .

En la práctica existen diferentes tipos de ROM, Estos se describen a continuación brevemente, la diferencia entre ellas se refiere principalmente a la manera en la que se obtienen las trayectorias de las funciones requeridas.

Las trayectorias requeridas en una ROM pueden programarse en dos formas diferentes. La primera se llama programación enmascarada y la hace el fabricante durante el último proceso de producción de la unidad. El fabricante hace la máscara correspondiente para las trayectorias para producir los 1 y 0 de acuerdo a una tabla de verdad.

Este procedimiento de programación enmascarada es bueno para grandes cantidades de ROM.

Para pequeñas cantidades, es más económico utilizar un segundo tipo de ROM llamado memoria programable de solo lectura (Programmable Read Only Memory) ó PROM. Cuando se ordenan, las unidades PROM contienen todos los 0 (ó los 1) en cada bit de las palabras almacenadas. Los eslabones en la PROM se rompen por la aplicación de pulsos de corriente a través de las terminales de entrada.



Un eslabón roto define un estado binario y un eslabón sin romper define el otro estado. Esto le permite al usuario programar la unidad en su propio laboratorio para lograr la relación deseada entre las direcciones de entrada y las palabras almacenadas. Están disponibles comercialmente unidades especiales denominadas programadores PROM para facilitar este procedimiento. En cualquier caso, todos los procedimientos para programar las ROM son procedimientos de hardware aun cuando se utilice la palabra programación.

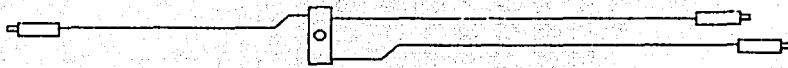
El procedimiento de hardware para programar las ROM ò PROM es irreversible y, una vez programados los patrones fijos, son permanentes y no pueden alterarse. Ya que se ha establecido un patrón de bits, la unidad debe descartarse si el patrón de bits va a cambiarse.

Un tercer tipo de unidad disponible se conoce como PROM borrable o EPROM. En las EPROM puede reestructurarse todo a su valor inicial (todos 0 o todos 1) aun cuando se hayan cambiado previamente. Cuando una EPROM se coloca bajo una luz ultravioleta especial por un periodo dado de tiempo, la radiación de onda corta descarga las compuertas internas que sirven como contactos. Después del borrado, la ROM regresa a su estado inicial y puede reprogramarse. Ciertas ROM pueden borrarse con señales eléctricas en lugar de luz ultravioleta, y se llaman EEPROM.

3.3.6. - Arreglo lógico programable PLA.

Ocasionalmente es posible que un circuito combinacional tenga condiciones de diseño no importa. Cuando se implementa con una ROM, una condición no importa se vuelve una entrada de dirección que nunca ocurrirá. Las palabras en las direcciones no importa no necesitan programarse y pueden dejarse en su estado original (todas 0 ò todas 1). El resultado es que no se usan todos los patrones de bit disponibles en la ROM, lo cual puede considerarse un desperdicio de equipo disponible.

Para casos donde el número de las condiciones no importa es excesivo, es más económico usar un segundo tipo de componente LSI llamado arreglo lógico programable ò PLA. Un PLA es similar en concepto a una ROM; Sin embargo, el PLA no proporciona la plena decodificación de las variables y no genera todos los términos como en la ROM. En el PLA, el decodificador se reemplaza por un grupo de compuertas AND, cada una de las cuales puede programarse para generar un término producto de las variables de entrada. Las compuertas AND y OR dentro del PLA están fabricadas inicialmente con eslabones entre ellas. Las funciones booleanas específicas se implementan en la forma de suma de productos por la apertura de los eslabones apropiados y dejando las conexiones deseadas.



El tamaño del PLA se especifica por el número de entradas. El número de términos producto y el número de salidas (el número de términos suma es igual al número de salida). Un PLA típico tiene 16 entradas, 48 términos producto y 8 salidas (circuito integrado TTL 82s100).

Como una ROM, el PLA puede ser programable por máscara o programable en campo.

El PLA que se conoce como arreglo lógico programable en campo o FPLA, puede programarlo el usuario mediante ciertos procedimientos recomendados. Existen disponibles unidades comerciales de hardware programable para usarse junto con ciertos FPLA.



3.4. - Microcontroladores.

Previo a la definición de un microcontrolador, explicaremos en qué consiste un microprocesador, puesto que es común el que haya confusión entre uno y otro nombre.

Un microprocesador es un circuito combinacional y secuencial. Es a su vez, un sistema digital síncrono de proceso. Esto significa que trabaja paso a paso realizando un proceso, activado por los impulsos de una señal de reloj y mediante una secuencia de operaciones que son indicadas por las instrucciones de un programa, el cual sigue un determinado diagrama de flujo.

Es digital, ya que las instrucciones, datos y direcciones de memoria son representados por cadenas de unos y ceros, lo que eléctricamente son voltajes altos y bajos.

Las funciones principales de un microprocesador son:

- Proveer las señales de tiempo y control para todos los elementos del sistema que lo rodea.
- Buscar instrucciones y datos desde la memoria.
- Transferir datos desde y hacia dispositivos de entrada / salida (E/S).
- Decodificar instrucciones.
- Realizar operaciones lógicas y aritméticas solicitadas a través de instrucciones.
- Responder a señales de control de E/S, tales como RESET o Interrupción.

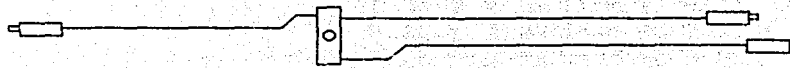
Un microcontrolador se compone de un microprocesador como elemento principal. Sin embargo, la diferencia entre ambos estriba en que el microcontrolador, aparte del microprocesador, tiene también espacio de memoria Ram, Rom y los buses de datos y direcciones.

Un bus de datos o direcciones, es un conjunto de canales (o cables) que se dedican a transportar la señal sea de datos o direcciones, como dice su nombre.

3.5. - Empleo del microcontrolador 87C51 de Intel.

El microcontrolador 87C51, pertenece a la familia de los '51 de Intel. Las principales características por las que se optó por su utilización son las siguientes:

- Tiene una memoria EPROM programable de 4k Bytes, y una RAM de 128 Bytes.
- Cuenta con una interfaz serial, compatible con una interfaz RS-232 programable.
- Es un microcontrolador de 8 bits.
- Maneja hasta 32 líneas como entrada o salida.
- Sus niveles de voltajes lógicos, son compatibles con tecnología TTL y CMOS.



El 87C51 cuenta con muchas otras características que a futuro dentro del mismo proyecto le hacen ser parte medular del sistema, por ejemplo, en caso de una expansión al sistema, se le puede instalar una memoria externa de 64k bytes de EPROM y con una lógica de TTL o CMOS, se le puede instalar un multiplexor, que incremente sus líneas de salida.

En la memoria EPROM interna del Microcontrolador o Micro, se almacenara el programa fuente de la interfaz, cuyas funciones primordiales serán las siguientes:

- Habilitar la entrada de la señal de los sensores, accionando un buffer conectado a la salida de cada uno de los sensores.
- Codificar y almacenar dentro de su memoria RAM la información enviada por cada uno de los sensores.
- Enviar a la computadora los datos recogidos por la codificación de la señal de los Sensores.
- Recibir de la computadora señales de control para inicio y paro de actividades de Medición.

El 87C51 cuenta con tres modos de operación. Según el modo en que se encuentre se deberán configurar algunas terminales de manera diferente y también cabe aclarar que existe un cambio en la función de algunas terminales e incluso también el consumo de corriente varía en el Micro.

Los modos de operación son los siguientes:

1. - Modo Activo (Active Mode).
2. - Único (Once).
3. - Potencia Baja (Power Down).

El modo Único se utiliza para efectuar pruebas a algunos subsistemas del micro, el modo de potencia baja se utiliza en aplicaciones donde el micro se lleve en una tableta portátil, que opere con baterías y se caracteriza por utilizar un mínimo de funciones y de corriente hasta que al micro se le indique lo contrario.

En el Modo Activo, el micro tiene todas sus funciones disponibles, de hecho es el modo en que más se le utiliza, tal como se hará en este caso.

El diagrama de conexión de las señales de habilitación para el Micro en Modo Activo es el que sigue:

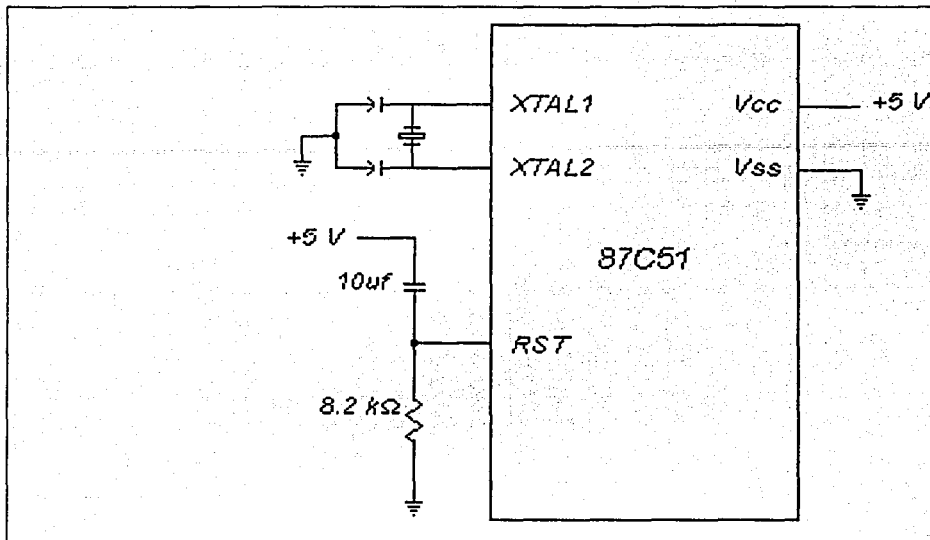
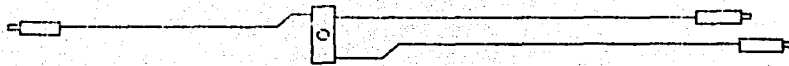


Figura 3.6.

Diagrama de Conexión del Reloj del Microcontrolador 87C51

3.5.1. - Programa y Diagrama de Flujo del Microcontrolador.

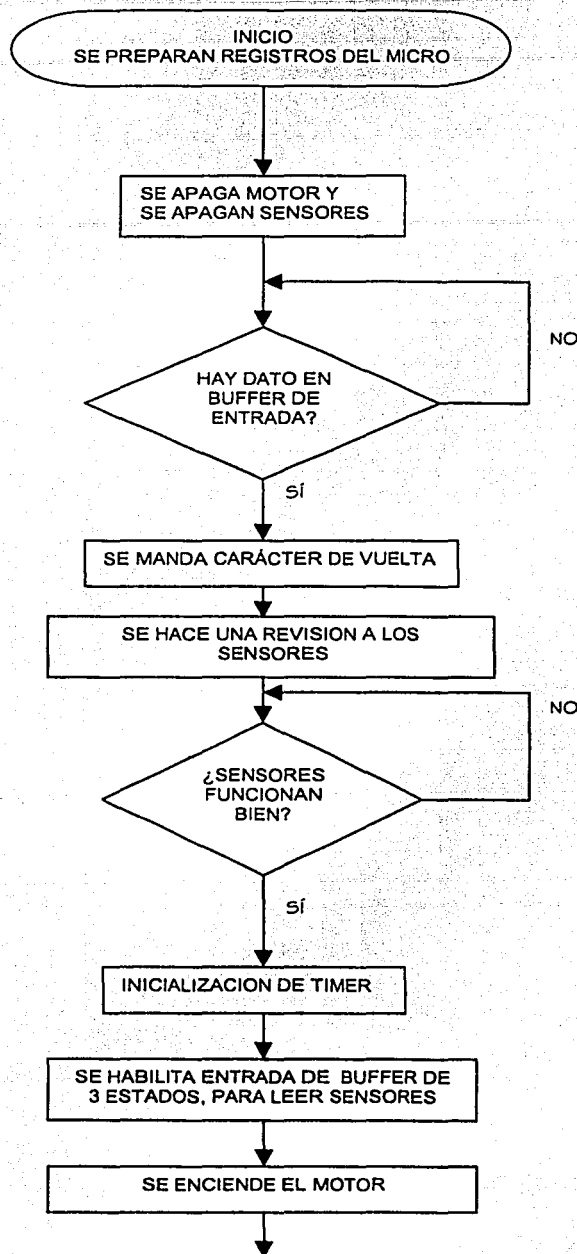
Dentro de los alcances del programa del Microcontrolador, se encuentran las siguientes funciones:

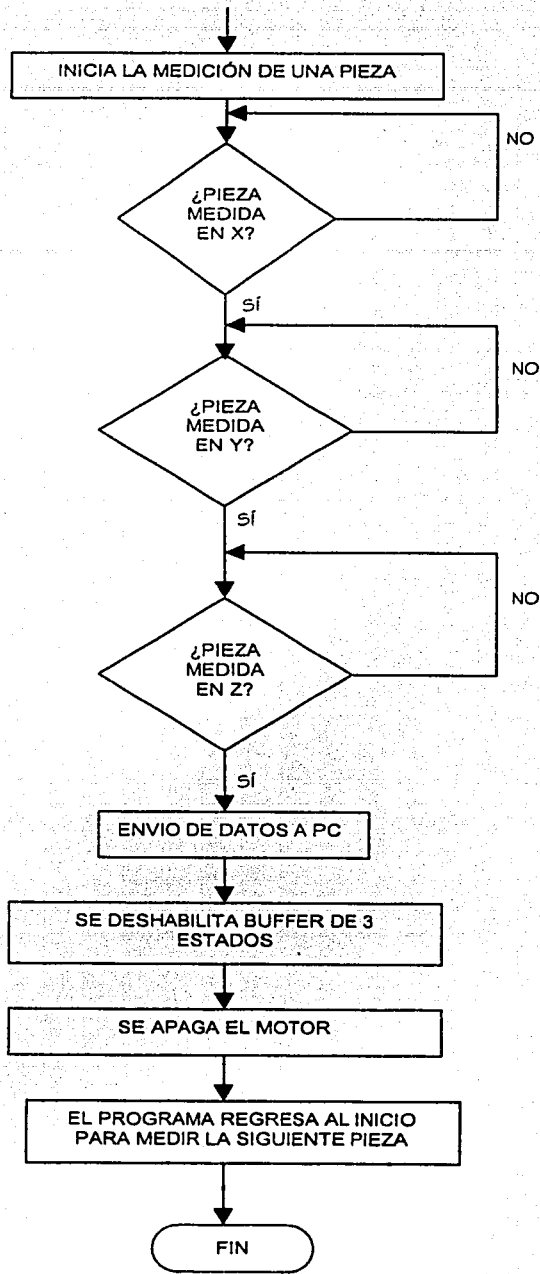
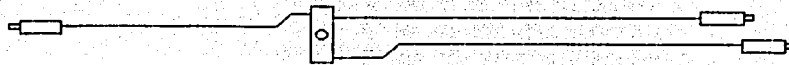
- Inicializar el puerto serial del Microcontrolador.
- Controlar el encendido y apagado del Motor.
- Controlar la habilitación o deshabilitación del buffer de tres estados, para recibir la información de los sensores.
- Interpretar las señales de voltaje que se reciban de los sensores y convertirlas en unidades de tiempo.
- Enviar la información obtenida de las mediciones a la Computadora Personal
- Avisar si algún sensor esta obstruido o apagado.

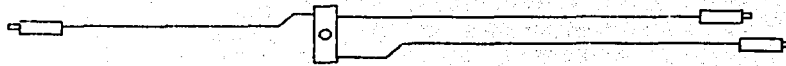
Para observar con más detalle como opera este programa a continuación se presenta el diagrama de flujo, indicando el llamado a las principales rutinas y sus respectivos saltos de programa. Más adelante se presenta el código fuente del programa en ensamblador del programa.



Diagrama de Flujo







Este diagrama de flujo indica el procedimiento que lleva el programa para operar. A continuación se presenta el código fuente del programa en lenguaje ensamblador para poder ver por completo todas las subrutinas.

Programa en Ensamblador

```

;*****
;          INICIALIZACION DEL MICRO Y SECCIONES DE MEMORIA
;=====
                .ORG H'0000          ; INICIALIZA LA RAM DEL MICRO
                LJMP H'0040          ; SALTA A LA DIRECCION 0040
INICIO:         .ORG H'0040          ; ENTRA A LA SECCION DE MEMORIA UVPRM
;*****
;          CUERPO PRINCIPAL DEL PROGRAMA
;=====

                MOV R0,#H'00          ; INICIALIZA EL REGISTRO 0
                MOV P1,#H'FF          ; DESHABILITA ENTRADA DE LOS SENSORES Y
MOTOR
                LCALL RETARDO          ; RUTINA QUE DA TIEMPO AL BUFFER
                LCALL TIMER1          ; LLAMA A HABILITACION DE TIMER1

AQUI:           LCALL RECIBE          ; RECIBE UN CARACTER EN SBUF -IN-
                MOV A,SBUF            ; EL CARACTER ES ESCRITO EN EL ACUMULADOR
                CLR ACC.7              ; DESENMASCARA EL 8vo. BIT (ASCII=7 BITS)
                LCALL MANDA           ; MANDA EL CARACTER DE VUELTA
                MOV P1,#H'FE          ; HABILITA ENTRADA DE SENSORES APAGA MOTOR
                LCALL RETARDO          ; RUTINA QUE DA TIEMPO AL BUFFER
                LCALL CHEQUEO          ; LLAMA A RUTINA DE CHEQUEO DE SENSORES
                LCALL MANDA           ; MANDA EL RESULTADO DEL CHEQUEO

NO_OK:         CJNE A,#H'57,NO_OK     ; SI FALLA UN SENSOR EL PROGRAMA PARA

                LCALL RETARDO          ; RUTINA QUE DA TIEMPO AL BUFFER
                LCALL PRENDE          ; LLAMA A RUTINA DE ENCENDIDO DEL MOTOR
                LCALL MEDICION         ; LLAMA A RUTINA DE MEDICION
                LCALL DATOS           ; LLAMA A RUTINA APARA ENVIO DE DATOS A PC
                LCALL RETARDO          ; RUTINA QUE DA TIEMPO AL BUFFER
                LCALL RETARDO          ; RUTINA QUE DA TIEMPO AL BUFFER
                LCALL RETARDO          ; RUTINA QUE DA TIEMPO AL BUFFER
                LCALL RETARDO          ; RUTINA QUE DA TIEMPO AL BUFFER
                LCALL APAGA           ; RUTINA PARA EL APAGADO DE MOTOR
                MOV P1,#H'FF          ; DESHABILITA ENT. DE LOS SENSORES Y MOTOR
                LCALL INICIAR          ; RUTINA QUE PREPARA EL REINICIO DEL MICRO
                SJMP AQUI              ; ETIQUETA DE FIN DE PROGRAMA PRINCIPAL

```



```

;*****
;      RUTINA DE RETARDO QUE PERMITE ACCIONAR AL BUFFER DE TRES ESTADOS
;=====
RETARDO:
MOV R7, #H'01      ; EL 74HC125 REQUIERE ___ms. PARA PRENDER
P1C:  MOV R6, #H'04 ; CON R0 Y R6 SE LE DA RETARDO DE 1 ms.
P2C:  MOV R0, #H'7D ; Y CON R2 SE LE DA AL RELEVADOR
P3C:  DJNZ R0, P3C   ; UN TIEMPO DE 1 ms. PARA APAGAR NORMAL
      DJNZ R6, P2C   ; DESPUES DE ESTE PASO, EL 74HC125 PUEDE SER
      DJNZ R7, P1C   ; ENCENDIDO OTRA VEZ SIN PROBLEMA
      RET            ; REGRESA A PROGRAMA PRINCIPAL

;*****
;      RUTINA QUE HABILITA AL TIMER1, PARA GENERAR 2400 BAUDS.
;=====
TIMER1:
MOV TMOD, #H'14    ; CONFIGURA TIMER 1 EN AUTO RELOAD-MODO2
MOV TCON, #H'02    ; CORRE TIMER 1 Y FIJA INTERRUPCIONES
MOV TH1, #H'005    ; CARGA TIMER PARA CORRER A 2400 BAUDS
MOV SCON, #H'3F    ; FIJA EL MODO 1 CON 8 BITS DE DATOS
RET                ; REGRESA A PROGRAMA PRINCIPAL

;*****
;      RUTINA QUE RECIBE CARACTER EN CODIGO ASCII
;=====
RECIBE:  JNB RI, RECIBE ; ESPERA AQUI HASTA RECIBIR UN CARACTER
         CLR RI        ; LIMPIA LA BANDERA DE RECEPCION
         RET           ; REGRESA A PROGRAMA PRINCIPAL

;*****
;      RUTINA QUE MANDA EL CARACTER DEL ACUMULADOR EN CODIGO ASCII
;=====
MANDA:   CLR TI        ; LIMPIA BANDERA DE TRANSMISION
         MOV SBUF, A   ; INICIA LA TRANSMISION CON EL ACUMULADOR

TXLOOP:  JNB TI, TXLOOP ; ESPERA HASTA QUE EL CARACTER SEA ENVIADO
         RET           ; REGRESA A PROGRAMA PRINCIPAL

;*****
;      RUTINA QUE VERIFICA EL FUNCIONAMIENTO DE LOS SENSORES
;=====
CHEQUEO: JB P2.0, SENSOR1 ; BRINCA SOLO SI HAY ERROR EN SENSOR 1
         JB P2.1, SENSOR2 ; BRINCA SOLO SI HAY ERROR EN SENSOR 2
         JB P2.2, SENSOR3 ; BRINCA SOLO SI HAY ERROR EN SENSOR 3
         MOV A, #H'57     ; MANDA UNA "W" SI ESTA TODO BIEN
         RET             ; REGRESA A PROGRAMA PRINCIPAL

SENSOR1: MOV A, #H'58     ; MANDA UNA "X" PARA ERROR EN SENSOR 1
         RET             ; REGRESA A PROGRAMA PRINCIPAL

```



```

SENSOR2:  MOV A, #H'59          ; MANDA UNA "Y" PARA ERROR EN SENSOR 2
          RET                   ; REGRESA A PROGRAMA PRINCIPAL

SENSOR3:  MOV A, #H'5A          ; MANDA UNA "Z" PARA ERROR EN SENSOR 3
          RET                   ; REGRESA A PROGRAMA PRINCIPAL

;*****
;  RUTINA QUE ENCIENDE AL MOTOR, CONSIDERANDO EL ENCENDIDO DEL RELE
;*****
=====

PRENDE:
          MOV P1, #H'FC         ; HABILITA ENTRADA DE SENSORES PRENDE MOTOR
          MOV R3, #H'19         ; EL RELEVADOR REQUIERE DE UN TIEMPO MINIMO
P1A:      MOV R2, #H'0F         ; DE 15 ms. PARA ENCENDER Y QUE LAS PLACAS
P2A:      MOV R1, #H'04         ; QUE VAN A LA BOBINA CIERREN
P3A:      MOV R0, #H'7D         ; R0 Y R1 GENERAN UN RETARDO DE 1 ms.
P4A:      DJNZ R0, P4A          ; CON R2 EL CICLO SE REPITE 16 VECES
          DJNZ R1, P3A          ; R3 NOS DA UN TIEMPO EXTRA PARA ASEGURAR
          DJNZ R2, P2A          ; EL FUNCIONAMIENTO DEL RELEVADOR DESPUES
          DJNZ R3, P1A          ; DE ESTE PASO NO IMPORTA SI SE APAGA EL
          RET                   ; RELEVADOR, LAS PLACAS NO SE DAÑARAN

;*****
;  RUTINA QUE EFECTUA LA MEDICION DE LOS SENSORES
;*****
=====

MEDICION:
          ; MEDICION DE LONGITUD (SENSOR 1)

          MOV R0, #H'33         ; VALOR DE LA LONGITUD ESTARA EN R0
          MOV R3, #H'33         ; LOS DESBORDAMIENTOS DE R0 ESTARAN EN R3
PRUEBA1:  JNB P2.0, PRUEBA1    ; SE QUEDA AQUI HASTA DETECTAR UNA PIEZA

UNO:      MOV R6, #H'FF         ; VALOR CON QUE SE CARGA EL CONT. EXTERNO
          MOV R7, #H'FF         ; AQUI SE CARGA EL VALOR DEL CONT. INTERNO
          INC R0                 ; SE INCREMENTA EL VALOR DE LA MEDICION

          CJNE R0, #H'FF, CICLO1 ; RUTINA PARA CORREGIR DESBORDAMIENTO DE R0
          INC R3                 ; SE INCREMENTA EL CONT. DE DESBORDAMIENTOS
          MOV R0, #H'33         ; SE REINICIALIZA R0

CICLO1:   DJNZ R7, CICLO1       ; REDUCE UN NUMERO DEL CICLO INTERNO
CICLO2:   DJNZ R6, CICLO2       ; REDUCE UN NUMERO DEL CICLO EXTERNO
          JB P2.0, UNO          ; PERMANECE EN RUTINA "UNO" HASTA MEDIR -X-

          ; MEDICION DE ANCHO (SENSOR 2)
          MOV R1, #H'33         ; VALOR DEL ANCHO ESTARA EN R1
          MOV R4, #H'33         ; LOS DESBORDAMIENTOS DE R1 ESTARAN EN R4
PRUEBA2:  JNB P2.1, PRUEBA2    ; SE QUEDA AQUI HASTA DETECTAR UNA PIEZA

DOS:      MOV R6, #H'FF         ; VALOR CON QUE SE CARGA EL CONT. EXTERNO
          MOV R7, #H'FF         ; AQUI SE CARGA EL VALOR DEL CONT. INTERNO
          INC R1                 ; SE INCREMENTA EL VALOR DE LA MEDICION

```

```

CJNE R1, #H'FF, CICLO3 ; RUTINA PARA CORREGIR DESBORDAMIENTO DE R1
INC R4 ; SE INCREMENTA EL CONT. DE DESBORDAMIENTOS
MOV R1, #H'33 ; SÉ REINICIALIZA R1

CICLO3: DJNZ R7, CICLO3 ; REDUCE UN NUMERO DEL CICLO INTERNO
CICLO4: DJNZ R6, CICLO4 ; REDUCE UN NUMERO DEL CICLO EXTERNO
JB P2.1, DOS ; PERMANECE EN RUTINA "DOS" HASTA MEDIR -Y-

; MEDICION DE ALTO (SENSOR 3)
MOV R2, #H'33 ; VALOR DE LA ALTURA ESTARA EN R2
MOV R5, #H'33 ; LOS DESBORDAMIENTOS DE R2 ESTARAN EN R5
PRUEBA3: JNB P2.2, PRUEBA3 ; SE QUEDA AQUI HASTA DETECTAR UNA PIEZA

TRES: MOV R6, #H'FF ; VALOR CON QUÉ SE CARGA EL CONT. EXTERNO
MOV R7, #H'FF ; AQUI SE CARGA EL VALOR DEL CONT. INTERNO
INC R2 ; SE INCREMENTA EL VALOR DE LA MEDICION

CJNE R2, #H'FF, CICLO5 ; RUTINA PARA CORREGIR DESBORDAMIENTO DE R2
INC R5 ; SE INCREMENTA EL CONT. DE DESBORDAMIENTOS
MOV R2, #H'33 ; SÉ REINICIALIZA R2

CICLO5: DJNZ R7, CICLO5 ; REDUCE UN NUMERO DEL CICLO INTERNO
CICLO6: DJNZ R6, CICLO6 ; REDUCE UN NUMERO DEL CICLO EXTERNO
JB P2.2, TRES ; PERMANECE EN RUTINA "TRES" HASTA MEDIR-Z-
RET ; REGRESA A PROGRAMA PRINCIPAL

```

```

;*****
; RUTINA QUÉ ENVIA LOS DATOS DE LA MEDICION A LA COMPUTADORA
;*****
=====

```

```

DATOS: ; RUTINA QUÉ ENVIA LOS DATOS DE LA MEDICION
MOV A, #H'40 ; CARGA UNA -@- EN ACUMULADOR PARA INICIAR
LCALL MANDA ; LA TRANSMISION DE LAS TRES MEDICIONES

MOV A, R0 ; CARGA EL VALOR DE LA LONG. EN ACUMULADOR
LCALL MANDA ; MANDA EL VALOR DE LA LONG. EN HEXADECIMAL
MOV A, R3 ; CARGA EL VALOR DEL DESBORDAMIENTO DE R0
LCALL MANDA ; MANDA EL VALOR DEL DESBORDAMIENTO DE R0
MOV A, R1 ; CARGA EL VALOR DEL ANCHO EN ACUMULADOR
LCALL MANDA ; MANDA EL VALOR DEL ANCHO EN HEXADECIMAL
MOV A, R4 ; CARGA EL VALOR DEL DESBORDAMIENTO DE R1
LCALL MANDA ; MANDA EL VALOR DEL DESBORDAMIENTO DE R1
MOV A, R2 ; CARGA EL VALOR DEL ALTO EN ACUMULADOR
LCALL MANDA ; MANDA EL VALOR DEL ALTO EN HEXADECIMAL
MOV A, R5 ; CARGA EL VALOR DEL DESBORDAMIENTO DE R2
LCALL MANDA ; MANDA EL VALOR DEL DESBORDAMIENTO DE R2
RET

```

```

;*****
; RUTINA QUÉ APAGA EL MOTOR
;*****
=====

```

```

APAGA: MOV P1, #H'FE ; HABILITA ENTRADA DE SENSORES APAGA MOTOR
MOV R2, #H'09 ; EL RELEVADOR REQUIERE DE 5 ms. PARA ABRIR

```



```

P1B:    MOV R1, #H'04      ; SUS PLACAS CON R0 Y R1 SE LE DA RETARDO
P2B:    MOV R0, #H'7D      ; DE 1 ms. Y CON R2 SE LE DA AL RELEVADOR
P3B:    DJNZ R0, P3B       ; UN TIEMPO DE 9 ms. PARA ABRIR NORMAL
        DJNZ R1, P2B       ; DESPUES DE ESTE PASO, EL MOTOR PUEDE SER
        DJNZ R2, P1B       ; ENCENDIDO OTRA VEZ SIN PROBLEMA
        RET

```

```

;*****
;          RUTINA QUE PREPARA LOS REGISTROS DEL MCU PARA INICIO
;=====

```

INICIAR:

```

CLR RI      ; LIMPIA LA BANDERA DE RECEPCION
CLR TI      ; LIMPIA LA BANDERA DE TRANSMISION
MOV A, #H'00 ; LIMPIA EL ACUMULADOR
MOV R0, #H'00 ; LIMPIA EL REGISTRO 0
MOV R1, #H'00 ; LIMPIA EL REGISTRO 1
MOV R2, #H'00 ; LIMPIA EL REGISTRO 2
MOV R3, #H'00 ; LIMPIA EL REGISTRO 3
MOV R4, #H'00 ; LIMPIA EL REGISTRO 4
MOV R5, #H'00 ; LIMPIA EL REGISTRO 5
MOV R6, #H'00 ; LIMPIA EL REGISTRO 6
MOV R7, #H'00 ; LIMPIA EL REGISTRO 7
RET

```

```

;*****
;          RUTINA QUE MARCA EL FIN DEL PROGRAMA
;=====

```

```

AJMP INICIO ; REGRESA CONTROL DEL PROGRAMA A INICIO
.END        ; FIN

```

```

;*****

```

A un lado de cada instrucción, el programa tiene escrito un comentario, esto se hace con el fin de conocer qué acción va ocurriendo en cada línea.



4.- PUERTOS DE UNA COMPUTADORA PERSONAL E INTERFAZ.

4.1. - Puertos típicos de una computadora personal.

Toda computadora personal tiene dentro de su arquitectura puertos de comunicación, estos puertos son diseñados para comunicar al CPU (Unidad Central de Proceso), con cualquier periférico externo que el operador requiera utilizar, siempre y cuando estos periféricos cuenten con una interfaz adecuada para la comunicación que han de llevar a cabo.

Entre mas especializado sea un modelo de computadora, nos encontraremos con puertos de las más diversas clases y funciones.

Por ello nos centraremos en los puertos más comunes para una computadora, de manera que el diseño sea compatible con todas las plataformas de computadora comercial disponible.

Dichos puertos son el Paralelo (LPT) y el puerto Serial (COM).

4.2. - Modo de operación de un puerto paralelo.

Un puerto paralelo consta de 25 pines o terminales de conexión.

En sus 25 pines de señal, no sé esta considerando al chasis del puerto, en el cual se efectúa la conexión de tierra física.

Su principal característica y por la que recibe su nombre, se basa en su manejo de datos, es decir, que de manera simultánea envía todos sus datos de salida y a la vez recopila los datos de entrada simultáneamente dependiendo del direccionamiento que en ese momento tenga programado.

Aparte del manejo de datos que incluye las entradas y salidas, se lleva la señal de reloj, para sincronizar la comunicación entre los equipos.

La aplicación de un puerto paralelo se hace principalmente bajo las siguientes condiciones:

- Cuando se requiera de una velocidad de trabajo muy alta.
- Cuando la distancia entre los equipos de comunicación no sea mucha.

La primera condición se aplica para comunicaciones digitales con periféricos de 16 bits y la segunda a la economía en el armado del modelo al no utilizar tantas líneas de cable.



4.3. - Modo de operación de un puerto serie.

Un puerto serie consta de 9 pines o terminales de conexión, de igual manera que en el puerto paralelo, el puerto serie utiliza el chasis para la conexión de tierra física.

De sus terminales ocupa una para transmisión, otra para recepción y la de reloj. Este puerto es el más común para aplicaciones comerciales, dada su manera de operación.

En este caso el puerto también toma su nombre de su manera de operación, es decir, este puerto envía la información en serie un dato tras otro por la misma terminal y la recibe de la misma manera, un dato tras otro.

De hecho para los fines de este trabajo, se utilizará una interfaz conectada a un puerto serie, como se describe a continuación.

4.4. - Aplicación del puerto serie con una interfaz RS-232.

Se llama interfaz al circuito, conexiones y dispositivos que en conjunto hacen efectivo el intercambio de valores de voltaje y la sincronización del reloj (cuando la sincronización se requiere) entre una computadora y un dispositivo específico.

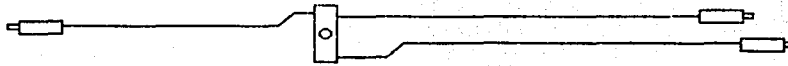
El nombre de interfaz RS-232 proviene del Circuito Integrado cambiador de voltajes, MAX RS-232, dicho cambiador de voltajes es un circuito integrado, cuya primordial función es la de adaptar los valores de los ceros y unos lógicos de la computadora a los de un circuito digital y viceversa.

Técnicamente la computadora para enviar un uno lógico envía 12 VOLTS y para un cero lógico envía -12 VOLTS.

En tanto que un circuito digital comúnmente utiliza 5 VOLTS para el uno lógico y 0 VOLTS para el cero lógico.

El MAX RS-232 se encarga exclusivamente de adaptar los valores de voltaje entre uno y otro dispositivo.

En lo referente a la sincronización de reloj, se utilizará para este caso un cristal de 8 Mhz, para las oscilaciones o pulsos de reloj del microcontrolador. A partir de este valor el microcontrolador ofrece una serie de valores en bauds (bits por segundo). En el programa interno del micro se especificará la velocidad de transmisión con la cual se desea establecer comunicación, a la computadora esto se notifica por medio del software en el que se programe.



Como paso final para establecer la comunicación, una vez que la computadora este ejecutando el programa fuente, se accionará el RESET del micro, para que al inicializar su funcionamiento, este automáticamente sincronice su señal de reloj, con la de la computadora.

A continuación se presenta el diagrama de operación del MAX RS-232

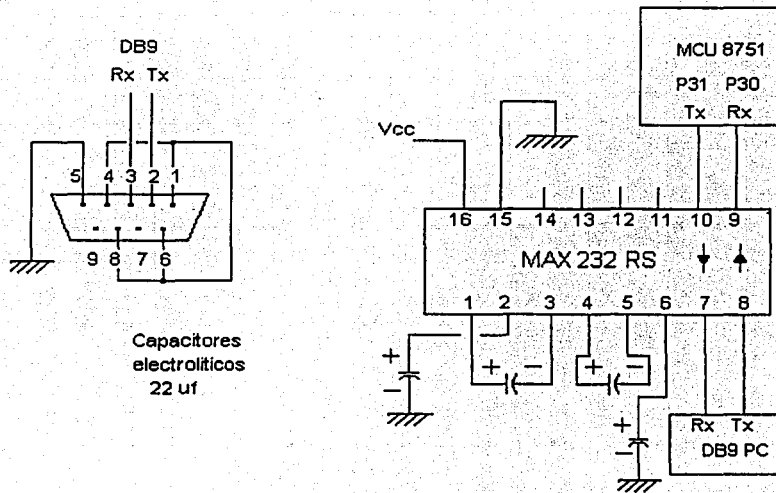
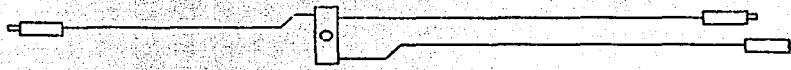


Figura 4.1
 Conexión del MAX RS - 232 a la Computadora Personal

El diagrama también describe la conexión de los pines en el conector DB 9 con el fin de que las terminales 1,4,5,6,8 cumplan el protocolo de comunicación sin problema.

TESIS CON
 FALLA DE ORIGEN



5.- DISTRIBUCIÓN DE LA BANDA TRANSPORTADORA.

El cuerpo de una banda transportadora es comercialmente conocido como transportador por lo cual le llamaremos de manera indistinta con estos dos nombres. El transportador se integra de 8 diferentes componentes.

Para efectos de este trabajo se requería de una banda transportadora que fuera a escala de las utilizadas en la industria ya que no se requiere de mucho espacio para este prototipo. Sin embargo, la banda transportadora comercial más pequeña mide 10 metros de largo y tiene por lo menos 1 metro de ancho.

En vista de lo anterior, se solicitó una cotización de una banda mucho menor en tamaño, que fuera funcional para nuestro propósito y conseguimos las siguientes especificaciones,

La longitud total del transportador era de 2.40 m. Por 0.58 m. De ancho total. La cama del transportador se conforma con una cama de rodillos de 2" de diámetro.

Sin embargo el costo de dicho transportador era algo elevado y sus dimensiones eran todavía muy grandes para este tema de tesis, por lo que finalmente se desarrolló un transportador acorde a las necesidades del prototipo.

5.1. - Especificaciones principales de la banda transportadora.

En este apartado se dará la descripción de los componentes básicos del transportador:

A) BASTIDOR.- hecho con ángulo de lámina galvanizada con perforaciones de 6.5 mm. A cada 5 cm. Con una longitud total de 51 cm. Y ancho total de 35.5 cm.

B) SOPORTES.- En Angulo de lámina galvanizada con perforaciones de 6.5 mm. A cada 5 cm. Con una altura de 27.5 cm. Con patín de goma antiderrapante y apoyos de tipo triangular para el bastidor.

C) POLEA MOTRIZ.- de resina plástica de 12.5 cm. De diámetro, con flecha circular de 8 mm. De diámetro.

D) RODILLO CARGADOR.- En tubo de PVC 38 mm. De diámetro, con flecha circular de 8 mm. De diámetro.

E) RODILLO DE RETORNO.- En tubo de PVC 38 mm. De diámetro, con balero y flecha circular de 8 mm de diámetro, colocados a 38 cms de distancia.



F) BANDA.- tela mezclilla de 25.5 cms de ancho, 1 capa color azul marino, de 42 cms de longitud.

G) TRANSMISIÓN.- Integrada por un motor de 1/25 HP monofásico a 127 VOLTS, de 1200 r.p.m. con juego de banda y polea de resina de 6 cm. De diámetro para una velocidad fija.

H) ACABADO.- el bastidor y los soportes tienen un terminado de pintura color negro. Y los apoyos de tipo triangular tienen un acabado galvanizado.

5.2.- Diagrama de la banda transportadora.

A continuación se presentan las vistas lateral y superior de las principales dimensiones del transportador con el fin de ubicar las características físicas con que se cuenta.

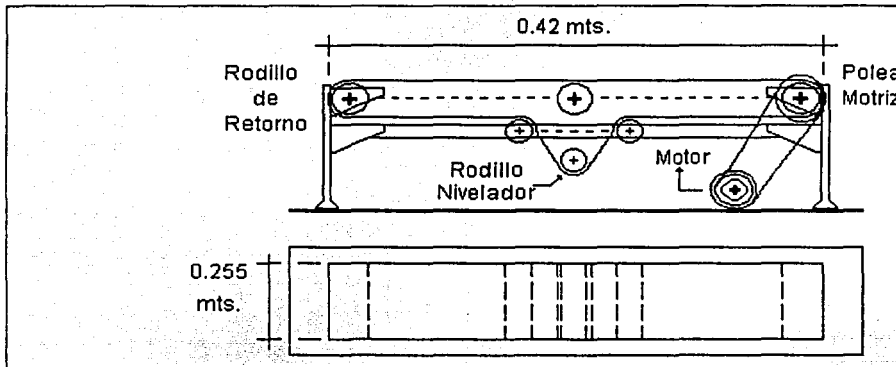


Figura 5.1.
Vista lateral y superior del transportador.

5.3. - Ubicación de los sensores en la banda transportadora.

La secuencia de medición se obtiene de la ubicación física de los sensores en el transportador. Esta fue la convencional X, Y y Z para generar la base de datos con el mismo orden.

En la siguiente figura podemos observar cual fue la disposición de los sensores dentro del transportador.

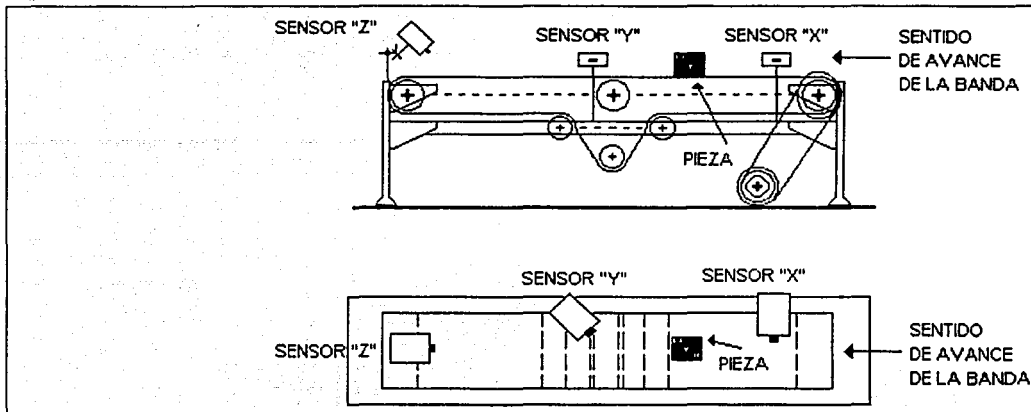
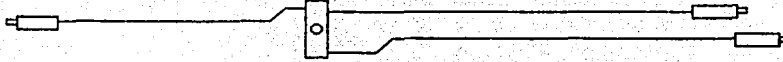


Figura 5.2

Ubicación de los sensores en la banda transportadora.

Como se pudo observar, los sensores X e Y van montados sobre el costado lateral de la banda. Variando solo por el ángulo en que ambos se encuentran instalados. El sensor X se instala perpendicular al avance de la pieza. El sensor Y se gira a 135° sobre la perpendicular que forma el sensor X en el sentido de avance de la banda.

Por otro lado, el sensor Z se encuentra montado en la parte posterior del transportador y genera la última medición. Para montarlo se requirió de un soporte especial que permita el paso de la pieza por debajo y la mida correctamente. Este sensor se encuentra a 135° con respecto a la vertical del plano como se ve en la Figura 5.2

5.4. - Circuito manejador del Motor de corriente alterna.

El motor de la banda transportadora es controlado por una pequeña interfaz de potencia, que lo enciende y apaga.

Esta etapa se subdivide en las siguientes fases:

- Entrada de señal para encendido y apagado.
- Optoacoplador para proteger a los semiconductores del Microcontrolador.
- Transistor de potencia media, para encender la bobina.

En la Figura 5.3 se verá gráficamente como opera este circuito manejador. Cabe



aclarar que la señal de voltaje que viene del Microcontrolador pasa antes de ingresar a la etapa de potencia por un Buffer de tres estados y aun así se decidió proteger a la circuitería electrónica con el Optoacoplador a la entrada del circuito de potencia, esto es con el fin de tener un sistema redundante en la protección de corriente para evitar eventualidades en los semiconductores.

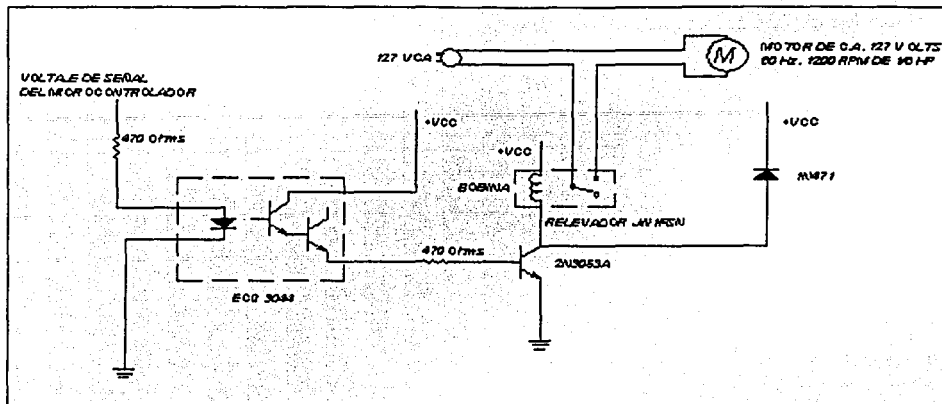
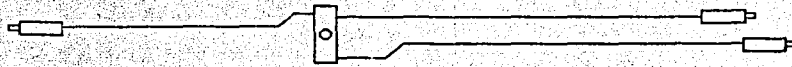


Figura 5.3
Diagrama de la etapa de potencia

En la Figura 5.3 se observa cómo la función de la etapa de potencia es solamente controlar el encendido y apagado del motor, controlando la fase. Cuando existe un voltaje lógico de 1 para habilitar el buffer de tres estados, de inmediato el diodo dentro del Optoacoplador prende la base del arreglo Darlington de base óptica y con ello se enciende a su vez el transistor de potencia media, quien al encender permite el flujo de corriente a través de su colector y con ello energiza la bobina del relevador que enciende y permite el paso de la fase de corriente alterna.

Cuando se envía un voltaje de uno lógico al buffer de tres estados este apaga el Led adentro del Optoacoplador y con el se apagan los transistores del arreglo Darlington y el de potencia media, desenergizando así la bobina, para contrarrestar el efecto inductivo que presenta una oposición al apagado, se coloca el diodo Damper, ese es el diodo 1N471, que esta al final del diagrama.



6.- BASE DE DATOS.

6.1. - Conceptos básicos.

La tecnología de las bases de datos se ha descrito como una de las áreas de la ciencia de la computación y de la información de más rápido desarrollo.

El campo rápidamente ha cobrado importancia práctica y teórica. La cantidad total de datos encomendados a las bases de datos se mide, sin exagerar, en varios miles de millones de *bytes*; la inversión financiera al respecto alcanza una cifra igualmente enorme; y no es exagerado afirmar que muchos miles de organizaciones dependen de la operación continua y eficaz de un sistema de base de datos.

Cuando se toma la tarea de definir qué es una base de datos, existen diversos conceptos, a través de los cuales la definición se puede expresar.

Aunque administración de una base de datos es un término informático, también se puede aplicar a las distintas formas en que se clasifica la información. En todo sistema de administración de información existe una base de datos.

Una definición de base de datos es la siguiente:

"Una base de datos es una colección organizada de información que guarda relaciones entre sí. Se agrupa integrando conjuntos y sirve para diferentes propósitos. La base de datos es la colección completa de datos, punteros, tablas, índices, etc."

Ahora bien, en esencia una base de datos, no es más que un sistema de mantenimiento de registros basado en computadoras, es decir, un sistema cuyo propósito general es registrar y mantener información.

Tal información puede estar relacionada con cualquier cosa que sea significativa para la organización del sistema donde opera.

Continuando con los conceptos básicos, daremos una pequeña definición de algunos factores que interactúan en el uso y desarrollo de una base de datos y que aplicaremos solo a ella.

Estos conceptos son: datos, hardware, software y usuarios.

- Datos.- Los términos "datos" e "información" se utilizarán en este trabajo como sinónimos. En algunas referencias bibliográficas se hace una distinción entre ellos, se emplea el término "datos" para referirse a los valores registrados físicamente en la base de datos y el término "información" para aludir al significado de estos valores según el significado que les dé un usuario.

Para nosotros, la definición de "datos" será también para referirse a los valores registrados físicamente en la base de datos. Aunque también la llamemos "información" de manera indistinta.



- Hardware.- Para una base de datos, consideraremos el hardware exclusivamente como la unidad física donde la información será almacenada, y el concepto será igualmente válido para un diskette o para una unidad de disco duro.

- Software.- Entre la base de datos física en sí (es decir, el almacenamiento real de los datos) y los usuarios del sistema existe un nivel de software que a menudo recibe el nombre de sistema de administración de base de datos o DBMS (Database Management System)

Este maneja todas las solicitudes de acceso a la base de datos formuladas por los usuarios.

Una función general del DBMS, por tanto, es proteger a los usuarios de la base de datos contra los detalles a nivel hardware.

Dada la importancia del término, en el tema 6.2 se abarcarán los detalles de un DBMS.

- Usuarios.- Se consideran tres clases de usuarios. La primera la representa el programador de aplicaciones, encargado de escribir programas de aplicación que utilicen bases de datos.

La segunda clase de usuarios es entonces, el usuario final que acceda a una base de datos, a través de una terminal ya sea que conozca el lenguaje de aplicación, o que exista un operario que opere la terminal.

De cualquier manera, el usuario final puede realizar, en general, todas las funciones de recuperación, creación, supresión y modificación.

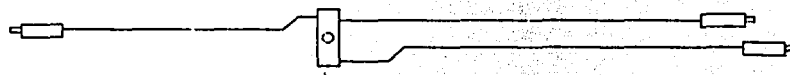
La tercera clase de usuario la representa el administrador de bases de datos o DBA él se encarga de optimizar los recursos del sistema, basándose en el ambiente que provee el primer usuario.

6.2. - Sistemas manejadores de base de datos.

Un sistema manejador de base de datos DBMS es un conjunto de programas especializados para describir, proteger, almacenar y acceder a una base de datos superando las limitaciones del procesamiento tradicional de archivos, garantizando la seguridad, integridad y protección de los datos, así como sincronizando el acceso múltiple de aplicaciones.

Por seguridad en ciertas aplicaciones, se entenderá que solo podrán acceder a la base de datos los usuarios autorizados y cada usuario autorizado verá sólo la parte que le corresponde. La integridad garantiza que los valores almacenados son correctos y que la base de datos es consistente. La protección asegura que en caso de *corrupción* de la base de datos, es posible recuperar los datos correctos.

Finalmente la sincronización controla que varias aplicaciones accedan en forma concurrente a la base de datos.



Si una base de datos se diseña, implementa y mantiene en forma correcta, un DBMS puede ayudar a una organización para aumentar su habilidad de responder a las necesidades cambiantes de información. Con la base de datos y el DBMS, el sistema de información se convierte en un sistema de base de datos esquematizado.

El sistema manejador de la base de datos se encarga de:

- La creación original de la descripción de la estructura de la base de datos y la forma en qué esta es reflejada por los archivos de la base de datos física.
- Establecer y mantener las trayectorias de acceso a la base de datos, de tal manera que los datos en cualquier parte de la base se puedan acceder rápidamente.
- Manejar los datos de acuerdo con las peticiones de los usuarios.
- Mantener la integridad y seguridad de los datos.
- Modificar la descripción de la base de datos, así como realizar copias de la misma.

El DBMS opera bajo tres niveles los cuales son: DDL, DML, y DQL.

El lenguaje de definición de datos (DDL), es utilizado por el administrador de la base de datos (DBA) para la creación de la base de datos, tablas, vistas y para establecer las medidas de seguridad en el acceso a las mismas.

El lenguaje de manipulación de datos (DML), permite al programador de aplicaciones consultar, insertar, actualizar la información y eliminar registros dentro de la base de datos.

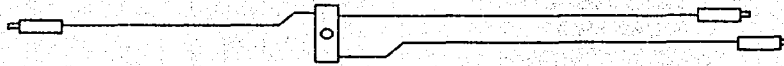
Y por ultimo el lenguaje de consulta de datos (DQL), es utilizado por el usuario final o casual para realizar consultas a la base de datos.

6.3. - Organización de la base de datos implementada.

La base de datos que se genera en archivo electrónico, es muy elemental para que pueda cumplir con los dos propósitos básicos con que fue diseñada, estos son:

- Ser compatible con la plataforma de Excel para Windows.
- Ser compatible con el software QUALITY CONTROL 4/e SQC 1.0 Student Versión, incluido en el libro Control De Calidad de Dale H. Besterfield.

Para lograr ambos propósitos, el Software denominado "WBAND1.EXE", genera un archivo de datos, con el nombre "medición.dat" este archivo es guardado en el mismo directorio en donde corre el Software.



A continuación, se verá un ejemplo del archivo "medición.dat" para comprender como esta organizada la información.

De esta manera se ve el archivo generado, cuando se observa desde un editor de texto:

| X | Y | Z |
|------|------|------|
| 3.28 | 4.17 | 5.10 |
| 5.23 | 2.42 | 3.69 |
| 4.24 | 4.24 | 4.24 |
| 3.47 | 5.62 | 2.93 |

En este ejemplo vemos cual es la organización básica del archivo, este siempre tendrá tres columnas y corresponderá cada una a la coordenada respectiva que estamos midiendo de la pieza en cuestión. El primer renglón corresponde a la literal de la coordenada que representa.

Abajo del renglón de literales vienen las mediciones correspondientes. En este caso se midieron 4 piezas y por ello hay cuatro renglones de datos.

El archivo "medición.dat" se sobrescribe en el disco duro, cuando se inicia una nueva sesión de mediciones. Si en alguna sesión se desea medir más piezas lo único que se debe hacer es colocar una nueva pieza en la banda y seleccionar la opción "CONTINUAR MEDICIÓN" y el archivo automáticamente ira añadiendo las mediciones tras la última efectuada, hasta el momento en que se termine la sesión y se cierre el software "WBANDI.EXE" en el sistema operativo Windows.

Cuando inicia una nueva sesión de medición con el programa "WBANDI.EXE" el archivo anterior de las mediciones, desaparece y es reemplazado por uno nuevo.

Para manejar el archivo desde la plataforma de Excel, se deberán seguir los siguientes pasos:

- Abrir el paquete de hoja de cálculo Excel.
- En la opción de abrir archivo, seleccionar el generado por la sesión previa "medición.dat" en el directorio donde se encuentre ubicado el programa "WBANDI.EXE"

NOTA: se deberá tener cuidado al seleccionar el archivo indicando la siguiente opción: -todos los archivos *.*- esto se hace para poder seleccionar el archivo en la ventana correspondiente a ABRIR ARCHIVO.

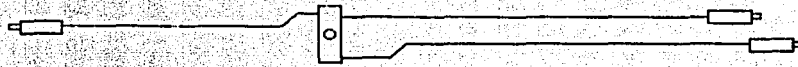
- Seguir el asistente indicado, para convertir este archivo de formato de datos, en una hoja de calculo del tipo ".xls".



- Con el archivo listo para trabajar en Excel, se puede utilizar la información capturada por la PC, de la manera normal con que se ocuparía en cualquier otro archivo de datos de Excel y aplicarle cualquier función estadística o gráfica, o incluso utilizarlo en algún archivo de Excel que contenga macros y que efectúe las operaciones que el alumno desee.

Para manejar el archivo desde la plataforma del Programa QUALITY CONTROL 4/e SQC 1.0 Student Versión, incluido en el libro Control De Calidad de Dale H. Besterfield, se deberán seguir los siguientes pasos:

- Abrir un editor de texto cualquiera, puede ser el editor incluido con el MS-DOS, o el NOTEPAD incluido en Windows o incluso el programa de Word.
- Dentro del editor de texto se deberá seleccionar la opción ABRIR ARCHIVO y entonces hay que seleccionar el subdirectorio donde corre nuestro software principal "WBAND1.EXE".
- En ese subdirectorio se debe dar la opción de abrir cualquier archivo (*.*) y entonces seleccionar "medición.dat" para abrirlo.
- Con el archivo abierto se deberá decidir con qué columna de información se desea trabajar, recordando que este Software está limitado a trabajar con solo una variable de calidad, que para estos efectos podrá ser X, Y o Z. Pero solo una a la vez. Si posteriormente se va a necesitar esa información con las tres variables juntas es conveniente renombrar el archivo y entonces editar el archivo renombrado como a continuación se explica. Se deberá borrar la información de las columnas con que no se va a trabajar desde el editor. También se deberán borrar los nombres de las literales X, Y y Z que solo sirvieron para efecto de identificar la información que recibimos de las mediciones.
- Una vez efectuado lo anterior se puede proceder a salvar los cambios en el archivo y cerrarlo.
- Con el archivo listo ahora si podemos entrar normalmente al Software de QUALITY CONTROL 4/e SQC 1.0 Student Versión, incluido en el libro Control De Calidad de Dale H. Besterfield Y trabajar con él.



7. - ANÁLISIS ESTADÍSTICO.

Para presentar los parámetros más utilizados de estadística, lo mejor es iniciar con la definición de esta rama de la ciencia.

7.1. - Definición de estadística.

En general son dos los significados que se asocian a la palabra estadística:

1. - Un conjunto de datos cuantitativos sobre un tema o grupo determinados, en especial cuando los datos se reúnen y agrupan de manera sistemática.
2. - La ciencia que se ocupa de la recopilación, tabulación, análisis, interpretación y presentación de datos cuantitativos.

Conviene destacar que este segundo significado es más general que el primero, dado que también se ocupa de la recopilación de los datos. La aplicación de la estadística en el control de la calidad esta relacionada con ese segundo significado, más amplio, e implica tareas como la recopilación, tabulación, análisis, interpretación y presentación de los datos cuantitativos.

Existen dos clases de estadística:

1. - *Estadística descriptiva o deductiva*, que se ocupa de la descripción y análisis de un tema o grupo.
2. - *Estadística inductiva*, cuyo objeto es, a partir de una determinada cantidad de datos (muestra), obtener una conclusión importante acerca de una mayor cantidad de datos (población). Dado que no es posible establecer conclusiones o inferencias con total certeza, se utilizan términos del lenguaje de la probabilidad.

7.2. - Elementos de estadística.

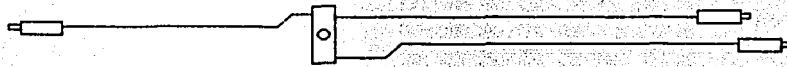
- Recopilación de datos.

La recopilación de datos se realiza mediante la observación directa o indirecta, a través de preguntas hechas por escrito o verbalmente.

La recopilación de datos que está enfocada al uso en el control de la calidad, se obtiene mediante observación directa y se clasifica como variables o como atributos.

7.2.1. - Datos no agrupados.

Los datos no agrupados son un listado de valores observados. Los datos agrupados representan el agrupamiento de todos los valores observados. Los datos pueden ser discretos, o continuos.



Dado que con datos desorganizados no se puede obtener información alguna, es necesario contar con un método que permita organizarlos.

El primer paso consiste en hacer un ordenamiento, es decir, organizar los datos numéricos desordenados por orden ascendente o descendente de magnitud.

El siguiente paso consiste en tabular la frecuencia con la que aparece cada uno de los valores. Como se muestra en la *Tabla 7.1.*

| Número de rechazos. | Tabulación. | Frecuencia. |
|---------------------|-------------|-------------|
| 0 | | 9 |
| 1 | | 13 |
| 2 | | 5 |
| 3 | | 4 |
| 4 | | 3 |
| 5 | | 1 |

Tabla 7.1.

Ejemplo para obtener la frecuencia en que se repite un evento, (en este caso son rechazos de una fábrica)

Si se eliminara la columna correspondiente a "Tabulación", la tabla resultante se conoce como *distribución de frecuencia*, y viene a ser un ordenamiento de los datos que permite observar los valores de frecuencia de cada una de las categorías.

La distribución de frecuencia es un método muy útil para visualizar datos y es un concepto fundamental de estadística.

Para mayor cantidad visual, las distribuciones de frecuencia se presentan en forma gráfica. Existen diversas formas de presentar las frecuencias de una distribución.

El histograma esta formado por un conjunto de rectángulos que representan la frecuencia de cada categoría. Representa gráficamente las frecuencias correspondientes a los valores observados. La *Figura 7.1.* es un histograma de los datos de la *Tabla 7.1.*

Dado que se trata de una variable discreta, teóricamente podría haberse representado utilizando una línea vertical en vez de un rectángulo. Sin embargo, esto es lo más convencional.

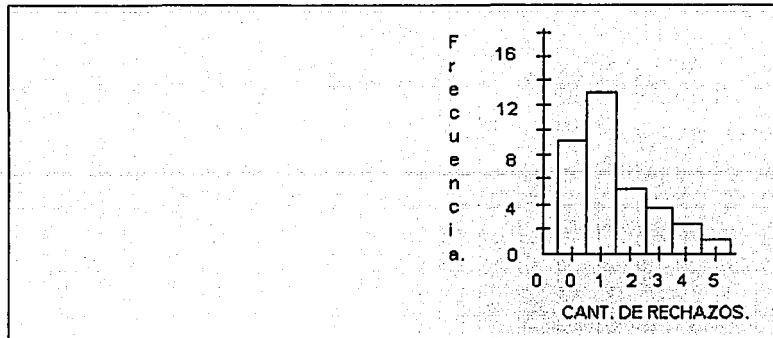


Figura 7.1.
Histograma de Frecuencias

Este diagrama se conoce como Histograma de frecuencias, también existen los Histogramas de frecuencia relativa, frecuencia acumulativa y frecuencia relativa acumulada.

Como se observa en la Figura 7.2, todos los Histogramas grafican a la frecuencia en el eje de las ordenadas.

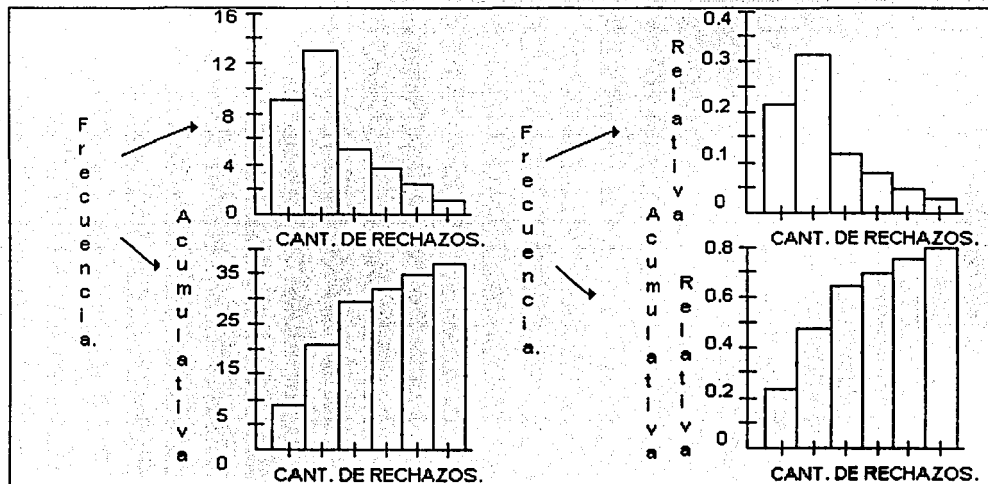


Figura 7.2.
Distintos tipos de Histogramas

TESIS CON
 FALLA DE ORIGEN

Para el eje de las abscisas, los valores que se grafican se obtienen de la siguiente manera:



- Frecuencia Relativa: Primero se obtiene la suma de todas las frecuencias y en cada punto a graficar de frecuencia, se dividirá entre la suma de las frecuencias, por ello se llama frecuencia relativa.

- Frecuencia Acumulativa: En este caso, se gráfica la primera frecuencia como en el caso normal, después al segundo valor de frecuencia se le sumará el valor de la primera, y en el caso de la tercer frecuencia se graficará la suma de los primeros dos valores sumados a este valor.

- Frecuencia Acumulativa Relativa: Este caso es una combinación de las dos técnicas anteriores. Inicialmente se obtiene la suma de todas las frecuencias, el primer valor a graficar será dividido entre el valor de la sumatoria de manera idéntica a la Frecuencia Relativa, en el segundo valor se deberá graficar la suma de los valores relativos de la primera y segunda frecuencia, y así sucesivamente para cada caso.

7.2.2. - Datos agrupados.

Construir la distribución de frecuencia de datos agrupados es más complicado debido a que por lo general hay más categorías.

Para conseguir la distribución de frecuencias en estos casos, se recurre a construir un histograma basado en celdas (ocasionalmente se utiliza la palabra clase en lugar de celda) formadas de subgrupos o intervalos, gráficamente las celdas y su nomenclatura se presentan a continuación:

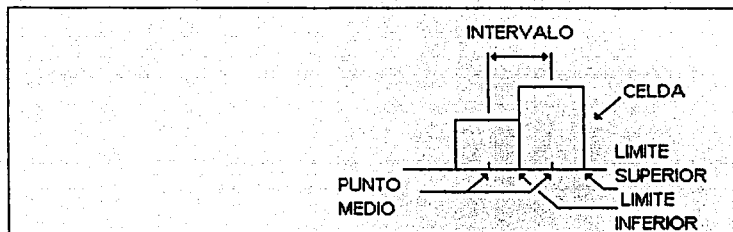
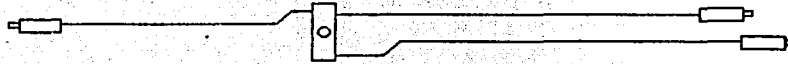


Figura 7.3.
Intervalo de una Celda

Los pasos a seguir para construir una distribución de frecuencias de datos agrupados son los que ahora se enlistan:

1. - Recopilación de datos y construcción de una hoja de anotaciones o marcas:

- Como primer punto se deberá observar cuantas variaciones de los datos nos encontramos en la muestra y el número de veces que un mismo valor se repita,



2. - Determinación del Rango:

- Para determinar el Rango, se buscará en la muestra el valor mayor que alcancen los datos, y le será restado el menor valor, al resultado de esta operación se le conoce como Rango de la muestra.

$$R = X_H - X_L$$

Donde

R = Rango.

X_H = número mayor.

X_L = número menor.

3. - Cálculo del intervalo de la celda:

El intervalo de una celda es la distancia que separa los puntos intermedios de dos celdas adyacentes, como se observa en la Figura 7.3. Es recomendable utilizar valores impares para los tamaños de la celda. (ejemplo 0.03, 0.009, etc.)

Cuando el valor de la celda es impar, se consigue que los puntos intermedios se encuentren a la misma distancia de lugares decimales que los valores de los datos.

El intervalo de la celda (i) y la cantidad de celdas (h) se relacionan mediante la fórmula

$$h = R / i.$$

Dado que tanto h como i se desconocen, se utiliza el método de ensayo y error para encontrar qué intervalo satisface las recomendaciones dadas.

4. - Cálculo de los puntos intermedios de las celdas:

El más pequeño de los puntos intermedios de las celdas deberá ubicarse de manera que se incluya en el valor más bajo de los datos. La técnica más sencilla consiste en escoger el punto más bajo de los datos como valor del punto intermedio de la primera celda. Una mejor técnica consiste en emplear la fórmula siguiente:

$$Mpl = X_L + (i/2) \quad \text{El resultado no se debe redondear.}$$

En donde Mpl = punto intermedio de la celda más baja.

5. - Cálculo de los límites de la celda:

Los límites de la celda son los valores extremo o límite de una celda, denominados límite superior y límite inferior.

Todas las observaciones cuyo valor quede comprendido entre los límites superior e inferior se clasifican dentro de la celda en particular. Los límites se definen de tal



manera que no quede duda de donde colocar el valor de una observación. Por lo tanto los valores de los límites cuentan con un valor decimal o figura significativa más de exactitud que los valores observados.

G. - Anotación de la frecuencia de una celda:

La cantidad de números presentes en cada celda se anota en la columna de frecuencias (o de valores repetidos) del punto 1.

La información acerca de la construcción de histogramas de frecuencia relativa, frecuencia acumulativa y frecuencia acumulativa relativa para datos agrupados es la misma que para datos no agrupados sin embargo existe una excepción. Con los dos histogramas de frecuencia acumulativa el límite verdadero de la celda superior es el valor que se identifica en la abscisa.

Aparte de los histogramas, existe otro tipo de gráficas para representar distribuciones de frecuencia, como se observa en la Figura 7.4.

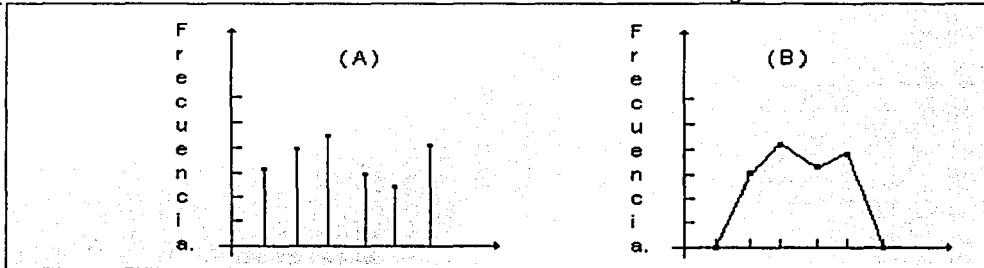


Figura 7.4.

Gráfico de barras de datos (A) y gráfico de polígono de datos (B).

Así también, se pueden identificar un comportamiento de frecuencia típico, solo con observar su diagrama, en la Figura 7.5. se muestran las más comunes.

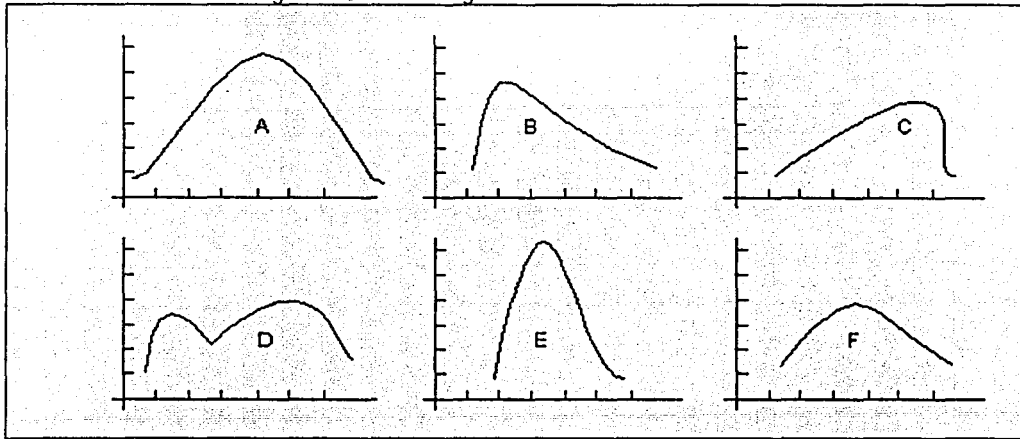


Figura 7.5.

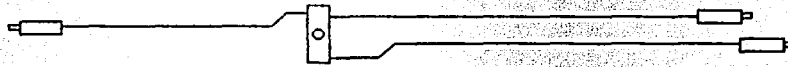


Gráfico de las distribuciones Típicas

Nombre de las distribuciones típicas:

- A - Distribución Simétrica o normal.
- B - Distribución con Asimetría o inclinación a la derecha.
- C.- Distribución con Asimetría o inclinación a la izquierda.
- D - Distribución Bimodal.
- E - Distribución Leptocurtica.
- F - Distribución Platicurtica.

7.3. - Medidas de tendencia central.

Para resolver muchos de los problemas del control de calidad bastará con recurrir a una distribución de frecuencia. Sin embargo existe un buen número de ellas en las que no se desee emplear una técnica gráfica, o para las que se necesite de la información adicional que proporcionan las técnicas analíticas.

Los métodos analíticos utilizados para describir un grupo de datos ofrecen la ventaja de requerir de menos espacio que una gráfica. También permiten comparar entre sí diversos grupos de datos. Asimismo permiten efectuar cálculos e inferencias adicionales. Son dos los métodos analíticos que más se utilizan para describir un grupo de datos: las medidas de tendencia central y las medidas de dispersión. Estas últimas se explicarán en la sección siguiente.

La medida de tendencia central de una distribución es un valor numérico que sirve para describir la ubicación central de los datos o en qué medida los datos tienden a agruparse en el centro. Se utilizan por lo general tres tipos de medidas: (1) La media, (2) La mediana y (3) La moda.

7.3.1. - La media.

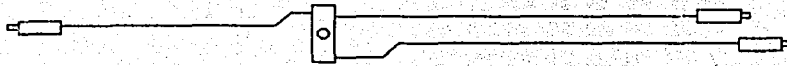
La media es la suma de las observaciones realizadas dividida entre la cantidad de observaciones correspondiente. Es la medida más común de una tendencia central. El cálculo de la media se puede hacer de tres formas: (1) de datos no agrupados, (2) de datos agrupados y (3) media ponderada.

1. - Técnica para datos no agrupados. Esta técnica se utiliza cuando los datos no están organizados.

La media se representa por la notación \bar{X} , que se lee como "x testada", y se calcula mediante la fórmula:

$$\bar{X} = (\sum_{i=1}^n x_i) / n = (x_1 + x_2 + \dots + x_n) / n$$

Donde :



\bar{X} = media

n = cantidad de valores observados.

x_1, x_2, \dots, x_n = valor observado que se identifica mediante el subíndice.

Σ = símbolo que indica la sumatoria o suma de valores (la suma inicia desde el valor del subíndice, y termina en el n - esimo valor, por lo general)

2. - Datos agrupados. Una vez que se agrupan los datos de conformidad con una distribución de frecuencia, se puede utilizar la siguiente técnica. La fórmula para obtener la media de los datos agrupados es:

$$\bar{X} = (\sum_{i=1}^n f_i x_i) / n = (f_1 x_1 + f_2 x_2 + \dots + f_n x_n) / (f_1 + f_2 + \dots + f_n)$$

Donde :

n = suma de frecuencias

f_i = frecuencia de una celda o frecuencia de un valor observado

x_i = punto intermedio de una celda o de un valor observado

h = cantidad de celdas o de valores observados.

Esta fórmula se utiliza cuando se trata de agrupamientos de celdas en donde hay más de un valor observado, por celda.

3. - Media ponderada. Cuando varias medias se combinan con distintas frecuencias, se está calculando una media ponderada.

La media ponderada se calcula utilizando la sig. Fórmula:

$$\bar{X}_w = (\sum_{i=1}^n w_i x_i) / \sum_{i=1}^n w_i$$

Donde :

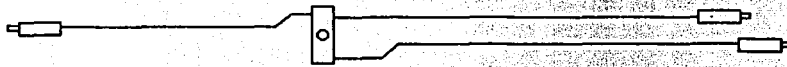
\bar{X}_w = media ponderada.

w_i = peso de la media.

7.3.2. - La mediana.

Otro valor que sirve para medir la tendencia central es la mediana, y se define como el valor que divide una serie de observaciones ordenadas de manera tal que la cantidad de elementos que la precede es igual a la cantidad de elementos que la siguen.

1. - Técnica para datos no agrupados. Hay dos situaciones que se pueden presentar al calcular la mediana de una serie de datos no agrupados: cuando la



cantidad de la serie es impar y cuando es par. En el primer caso, la mediana corresponde al punto medio de los valores. Cuando la cantidad de elementos de la serie es par, la mediana corresponde al promedio de los dos números de en medio. Siempre se deberá tener cuidado de que los números estén ordenados antes de proceder a calcular la mediana.

2. - Técnica para datos agrupados.

Si los datos están agrupados de conformidad con una distribución de frecuencia, la mediana se calcula ubicando la celda en donde esté el número de la posición media y luego haciendo una interpolación dentro de la celda. La fórmula de la interpolación para hacer este cálculo es la siguiente:

$$Md = L_m + \{[(n/2) - cf_m] / f_m\} i$$

Donde :

Md = mediana

L_m = límite inferior de la celda en donde esta la mediana

n = número total de observaciones.

cf_m = frecuencia acumulativa de todas las celdas abajo de L_m

f_m = frecuencia de la celda mediana.

i = intervalo de la celda.

7.3.3. - La moda.

La moda (M_o) de un conjunto de números es el valor que aparece con la mayor frecuencia. Puede suceder que en una serie no haya moda o que haya varias. Se dice que una serie de números es *unimodal* si solo tiene una moda, *bimodal* si tiene dos modas y *multimodal* si tiene más de dos modas.

Al agrupar los datos en una distribución de frecuencia el punto intermedio de la celda que tenga la frecuencia mayor corresponderá a la moda, dado que este punto representa al punto más alto (mayor frecuencia) del histograma. Para un mejor cálculo de la moda se efectúa una interpolación de manera similar a la utilizada para la mediana.

7.4. - Medidas de Dispersión.

En la sección anterior se comentó de las diversas técnicas utilizadas para describir la tendencia central de los datos. La segunda herramienta que se utiliza en la estadística consiste en lo que se denomina Medidas de dispersión. Mediante las que se describe como se diseminan o se dispersan los datos en torno del valor central.



Estas medidas de dispersión y las de tendencia central son necesarias para describir a un grupo de datos.

Las medidas de dispersión de las que se comentará en esta sección son el Rango y la Desviación estándar.

7.4.1. - Rango.

El rango de valores de una serie de números es la diferencia entre los valores u observaciones mayor o menor. Simbólicamente se expresa con la fórmula mostrada en la sección 7.2.2.

$$R = X_H - X_L$$

Donde :

R = Rango.

X_H = número mayor.

X_L = número menor.

El rango es la medida de dispersión más fácil y sencilla de calcular. Otra medida relacionada con la del rango, y que a veces se utiliza, es el *Rango medio*, es decir, el rango dividido entre 2, $R/2$.

7.4.2. - Desviación Estándar.

La desviación estándar es un valor numérico, expresado en las unidades de los valores observados, que mide la tendencia de dispersión de los datos. Una desviación estándar grande indica una mayor variabilidad en los datos que en el caso de una desviación estándar pequeña. Se expresa mediante la siguiente fórmula:

$$s = \left[\sum_{i=1}^n (x_i - \bar{x})^2 / (n - 1) \right]^{1/2}$$

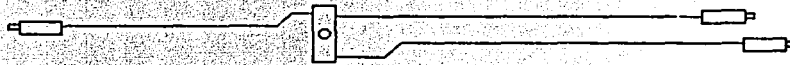
Donde :

\bar{x} = media

s = desviación estándar de la muestra

x_i = valor observado

n = cantidad de valores observados.



8. - CARACTERÍSTICAS DE CALIDAD.

8.1. - Introducción del término "Calidad"

Cuando se menciona el término "Calidad", por lo general lo asociamos con productos o servicios excelentes, que satisfacen nuestras expectativas y, más aún, las rebasan. Tales expectativas se definen en función del uso que se dará al producto o servicio en cuestión y de su respectivo precio de venta. Cuando un producto mejora nuestras expectativas estamos hablando de calidad. Es decir, se trata de una cualidad cuya valoración dependerá de lo que se perciba.

De acuerdo con la norma A3 -1987 ANSI/ASQC, calidad es la totalidad de aspectos y características de un producto o servicio que permiten satisfacer necesidades implícitas o explícitamente formuladas.

El control de la calidad es la aplicación de técnicas y esfuerzos para lograr, mantener y mejorar la calidad de un producto o de un servicio. Esto implica la integración de las técnicas y actividades siguientes relacionadas entre sí:

1. - Especificación de que se necesita.
2. - Diseño del producto o servicio de manera que cumpla con las especificaciones.
3. - Producción o instalación que cumpla cabalmente con las especificaciones.
4. - Inspección para cerciorarse del cumplimiento de las especificaciones.
5. - Revisión durante el uso a fin de allegarse información que, en caso de ser necesario, sirva como base para modificar las especificaciones.

8.2. - Variables y Atributos de calidad.

Las variables son aquellas características de la calidad en un producto (aplica en ocasiones a servicios también) que son medibles. Los atributos, por otra parte, son las características de la calidad y se dividen en dos grupos: los que satisfacen determinadas especificaciones y los que discrepan de ellas.

Toda variable susceptible de ser dividida infinitamente se denomina *continua*.

Las variables en donde hay vacíos o saltos se conocen como *discretas*.

En general, los datos continuos son medibles, en tanto que los discretos son resultado del conteo.

Hay ocasiones en que es necesario que los datos verbales o no numéricos puedan manejarse como variables. (Ejemplo: encuestas con calificación de excelente, bueno, regular, malo, etc.)

Si bien muchas de las características de la calidad se definen en función de variables, hay muchas otras que deberán definirse como atributos. Es muy frecuente que a aquellas características que se evalúan a través de una observación visual se les clasifique como atributos. (Ejemplo: este texto esta bien escrito ¿ o no?)



Al recopilar los datos, la cantidad de cifras de estos estará en función del uso que se les piense dar.

Cuando se trabaja con datos numéricos, es muy importante tomar en cuenta las cifras significativas. Las cifras significativas de un número son aquellas cifras con las que el número está escrito prescindiendo de los ceros que pueda haber a la izquierda.

En el caso de las operaciones matemáticas de multiplicación, división, potenciación y radicación, el resultado obtenido tendrá la misma cantidad de cifras significativas que el número que figure en la operación y que tenga la mínima cantidad de cifras significativas.

8.3. - Control Estadístico del Proceso.

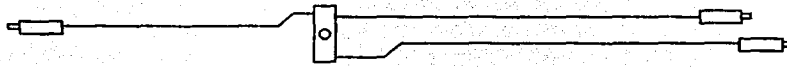
El control estadístico de la calidad (SQC, statistical quality control) es una rama del control de la calidad. Consiste en el acopio, análisis e interpretación de datos para su uso en el control de la calidad.

El control estadístico del proceso y muestreo de aceptación son dos de los más importantes elementos del control estadístico de la calidad. Sin embargo, son muchas las técnicas de las que hay que echar mano.

Uno de los axiomas o verdades evidentes de la fabricación en cualquier escala, es que nunca se producen dos objetos exactamente iguales. En realidad esta idea de la variación es una ley inherente a la naturaleza, en donde no hay dos seres que pertenezcan a la misma categoría que sean exactamente iguales. La variación puede ser grande y evidente, como es el caso de la diferente altura de los seres humanos, o pequeña e imperceptible, como el peso de las puntas de los bolígrafos o la forma de los copos de nieve. Si las variaciones son muy pequeñas, los objetos darán la impresión de que son idénticos. Sin embargo, mediante instrumentos de precisión se hará patente la diferencia. Si dos objetos dan la impresión de medir lo mismo, se debe a las limitaciones de los instrumentos de medición empleados. Conforme se han ido mejorando estos, las diferencias han persistido, aunque el aumento en la variación ha cambiado. Antes de controlarla, hay que estar en condiciones para medir la variación, he ahí la importancia del control estadístico del proceso.

Existen tres clases de variación en la producción de las partes por pieza:

1. - Variación en la pieza misma. Ejemplo de este tipo de variación es el diferente grado de aspereza de la superficie de una pieza, en la que una área puede ser más áspera que otra; el ancho de uno de los extremos del ojo de una cerradura es distinto del otro extremo.



2. - Variación de una pieza a la otra. Este tipo de variación se produce en piezas que se fabrican al mismo tiempo. Por ejemplo, la distinta intensidad luminosa que producen cuatro focos fabricados por una máquina, uno después del otro.

3. - Variación de una hora a otra. Ejemplo de esta variación son las diferencias que hay en los productos fabricados a distinta hora del día. Lo que se fabrica en la mañana podrá ser diferente de lo que se hace más tarde; o conforme se desgastan las herramientas con que se manufactura, así también se modifican las características del producto final.

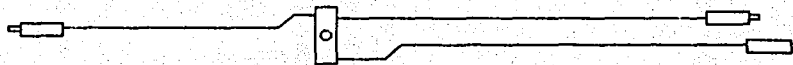
La variación es algo inherente a todo proceso, debido al efecto conjunto de *equipo, materiales, entorno y operario*.

- La primera causa de la variación es el *equipo*. En esta causa figuran el desgaste de la herramienta, las vibraciones de la máquina, el equipo de sujeción de trabajo y del posicionamiento de dispositivos así como las fluctuaciones hidráulicas y eléctricas. Cuando se conjuntan todas estas variaciones, el equipo operará dentro de cierta capacidad o precisión. Incluso se afirma que máquinas idénticas tienen capacidades diferentes, algo muy importante que se debe tomar en cuenta cuando se programe la fabricación de piezas importantes.

- La segunda causa de la variación es debido al *material*. Puesto que se producen variaciones en el producto terminado, también deben estar presentes en la materia prima (a su vez, otro producto terminado) Características relacionadas con la calidad como la resistencia a la tensión, ductilidad grosor, porosidad y contenido de humedad, es de esperar que contribuyan a la variación total del producto final.

- Una tercera causa de variación es el *entorno*. La temperatura, la luz, la radiación, el tamaño de las partículas, la presión y la humedad contribuyen todas a las variaciones en el producto. Para tener bajo control estos factores, a veces se fabrican los productos en habitaciones blancas. Hay experimentos que se realizan en el espacio exterior a fin de allegarse mayor información sobre el efecto del entorno en las variaciones del producto.

- Una cuarta causa de variación es el *operario*. En esta causa figura también el método que emplea el operario para realizar determinada operación. El bienestar emocional y físico del operario también contribuye a la variación. La falta de comprensión de un operario sobre las variaciones del equipo y del material debido a una falta de capacitación hará necesario efectuar continuos ajustes de máquina, con lo que la variabilidad se hace más compleja. Conforme nuestro equipo es más y más automatizado el efecto del operario en la variación del proceso ha disminuido.



8.3.1. - Método de la gráfica de control.

Para indicar cuando las variaciones que se registran en la calidad no rebasan el límite aceptable para el azar, se utiliza el método de análisis y presentación de datos conocido como método de la gráfica de control. Se trata de un registro gráfico de la calidad de una característica en particular. Muestra si un proceso está o no estable.

En la Figura 8.1. se muestra un ejemplo una gráfica de control, esta gráfica en particular se conoce como gráfica \bar{X} y en ella se registra la variación experimentada en el valor promedio de las muestras.

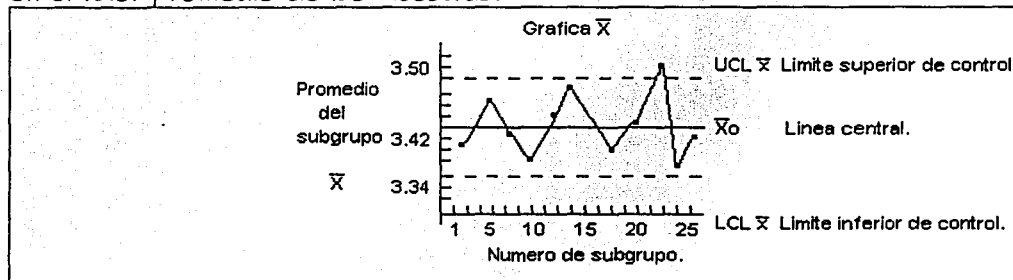


Figura 8.1.
Ejemplo de gráfica de Control

Al eje horizontal se le denomina "número correspondiente del subgrupo", mediante el que se identifica una muestra en particular formada por una cantidad fija de observaciones. Estos subgrupos aparecen por orden, el primero que se inspecciona es el 1 y el último es el 25.

El eje vertical de la gráfica corresponde a la variable, que puede ser representado por una longitud, peso, diámetro, rugosidad, densidad, etc.

Cada uno de los puntos representa el valor promedio de uno de los subgrupos, cada punto es un valor promedio. En las gráficas de control se usan los promedios en vez de observaciones individuales, dado que los valores promedio permiten mucho más rápido cualquier cambio en la variación.

Asimismo, contando con dos o más observaciones en una muestra, es posible tener una medida de dispersión de un determinado subgrupo.

La línea continua de la mitad de la gráfica se puede interpretar de tres maneras, dependiendo de los datos disponibles.



Primero, puede considerarse como el promedio de los puntos graficados. Segundo, se puede considerar como una norma o valor de referencia. Tercero, puede verse como la media de una población, si es que tal valor se conoce.

Las dos líneas de puntos son los límites de control superior e inferior. Estos límites son un auxiliar para juzgar el grado de variación que se produce en la calidad de un producto. Con frecuencia se confunden los límites de control con los límites de la especificación, siendo estos últimos los límites permisibles de una determinada característica de cada unidad particular de un producto. Por otra parte, los límites de control sirven para evaluar las variaciones producidas en la calidad de un subgrupo a otro. Por tanto, en el caso de la gráfica, los límites de control son función de los promedios de los subgrupos. Si se desea calcular la frecuencia de distribución de los promedios de los subgrupos se puede recurrir al promedio y la desviación estándar correspondientes. Los límites de control se fijan a ± 3 desviaciones estándar de la línea central.

8.4. - Interpretación de datos.

Una manera de interpretar los datos arrojados por una gráfica de control, se obtiene de analizar los valores estadísticos de la muestra y conjuntar lo anterior con los conceptos de población y muestra, para poder llegar a plantear ciertas inferencias.

8.4.1. - Concepto de población y muestra.

Al construir una distribución de frecuencia se escoge una pequeña porción o muestra representativa de todo el conjunto.

Por lo tanto es fácil identificar ambos conceptos y su diferencia. La población corresponde a la totalidad de los elementos del conjunto. Así, la muestra es una parte o subconjunto de la población.

Cuando se calculan las medias, desviaciones estándar y otro tipo de mediciones relacionados con las muestras, dichas medidas se denominan estadísticas. Dado que la composición de las muestras puede variar, las estadísticas calculadas resultaran ser mayores o menores que los valores reales de la población, y se conocen como parámetros. Los parámetros se consideran como valores de referencia fija (estándar) o como el cálculo más aproximado de estos valores en un momento determinado.

Dado que rara vez se puede medir la totalidad de la población, se opta por elegir una muestra. El muestreo es necesario cuando es imposible medir la totalidad de la población. Cuando el gasto que implica la observación de todos los datos es prohibitivo, cuando el proceso de revisión implica la destrucción del producto



cuando probar el comportamiento de la población entera podría ser muy peligroso, como en el caso de un nuevo fármaco. En realidad, el análisis de la población total puede no producir resultados más exactos que el del muestreo. Se ha demostrado que el realizar una revisión 100% en forma manual no dan resultados tan precisos como el muestreo. Quizás la razón sea que el aburrimiento y la fatiga favorecen que los inspectores prejuzguen cada elemento revisado considerándolo de antemano aceptable.

Se utilizan letras griegas para nombrar a las poblaciones. Como se verá a continuación en la Figura 8.2.

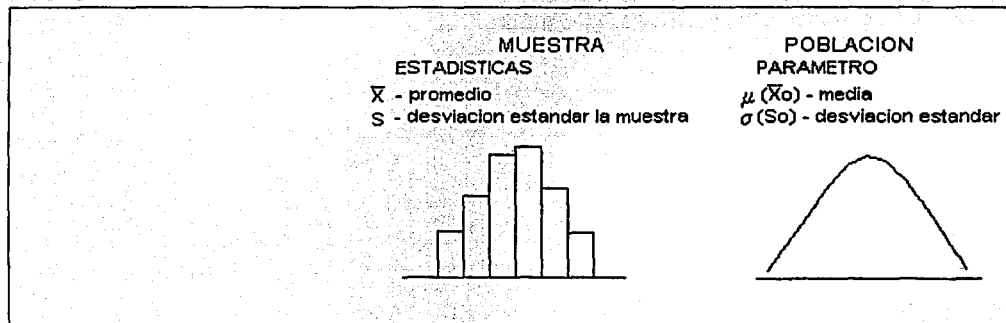


Figura 8.2.
Diagrama de Muestra y Población

El objetivo fundamental de elegir una muestra permite aprender algo sobre la población y sirve como auxiliar para tomar una decisión. La muestra escogida deberá ser tal que tienda a parecerse o representar a toda la población. Que tan bien representa la muestra a la población dependerá del tamaño de la muestra, del azar, del método de muestreo y del sí las condiciones cambian o no.



9.- CONCLUSIONES.

9.1.- Revisión de puntos originales.

Revisando los resultados obtenidos contra los alcances propuestos. Cotejaremos con los puntos iniciales para resumir las conclusiones.

- **SIMULACION DEL PROCESO INDUSTRIAL.-** Esta se logra con el prototipo a escala y entregando al usuario los controles de la banda así como permitiéndole ver las mediciones obtenidas en tiempo real. Hacerle conocer los requerimientos del sistema para operar es una buena manera de concientizarlo técnicamente de los cuidados que debe tener para su operación.
- **SENSORES.-** Los sensores cumplen con la primera etapa del prototipo porque obtienen de manera indirecta la medición de las longitudes deseadas. La medición se puede denominar indirecta, debido a que es a través del tiempo que tardan las piezas en pasar como se mide. Para desarrollos posteriores al prototipo, es probable que alguno o todos los sensores ópticos sean reemplazados por otro tipo de sensores. Para conocer de qué tipo de sensores estamos hablando, se deberá conocer la variable de calidad que se desea obtener en la base de datos final.
- **BASE DE DATOS.-** La base de datos puede ser leída en las dos aplicaciones que fueron requeridas en los requerimientos del trabajo a desarrollar y estas son: El Software de QUALITY CONTROL 4/e SQC 1.0 Student Versión, incluido en el libro Control De Calidad de Dale H. Besterfield. Para el que se deberá de editar la dimensión con que se quiere operar el software, tal como se explica en el capítulo 6 de la base de datos.

El otro software es el conocido Excel de Microsoft Office. En este caso se puede abrir con el asistente incluido en Excel para convertir esa base de datos en una hoja de cálculo de Excel.

**ESTA TESIS NO SALI
DE LA BIBLIOTECA**



- CONTROL DE PROCESO EN PC.- El control de la computadora personal en esta fase del prototipo es un control compartido ya que la computadora personal no realiza todas las funciones y tampoco el microcontrolador 87C51 tiene esa capacidad. Por ello se dice que el control de las funciones está compartido entre ambos elementos y ambos se apoyan en su software para dicho fin. El control del proceso a través del prototipo comprende los siguientes tópicos:
 - a) Encender el motor del transportador.
 - b) Apagar el motor del transportador.
 - c) Avisar si existe un error en los sensores.
 - d) Identificar qué sensor esta fallando.
 - e) Proteger al microcontrolador a través de un buffer de tres estados.
 - f) Generar la base de datos.
 - g) Guiar al usuario a través de la interfaz gráfica.

9.2.- Expansiones a futuro para el prototipo.

Con todos los elementos arriba indicados se demuestra que los alcances del tema fueron cabalmente cumplidos y además se queda una infraestructura sólida en los laboratorios de la Facultad. Para generar las etapas posteriores en las cuales podrán, entre otras, efectuar las siguientes modificaciones al sistema del transportador:

- Cambiar el tipo de sensores por unos distintos, según se requiera en las prácticas de laboratorio.
- Cambiar el motor por uno de corriente directa que opere con pulsos para poder controlar la velocidad de operación así como el sentido mismo de avance de la banda.
- Crear un software programado ex profeso para leer la información de la base de datos generada por el prototipo. Esto se puede hacer en un ambiente gráfico para facilitar su uso y aplicación.
- Escalar el software de versión para utilizar el prototipo en los sistemas operativos modernos como Windows 95, Windows 95 Plus, Windows 98, Windows NT 4.0, Windows ME, entre otras versiones de dicho sistema operativo. Esta versión fue desarrollada para plataforma de Windows 3.1 como un requerimiento de funcionamiento por la disponibilidad de los equipos en el laboratorio.



APÉNDICE A.- FUENTE DE ALIMENTACIÓN.

Para facilitar los voltajes de alimentación del sistema se estandarizó todo el sistema a 5 Volts de corriente directa.

Se escogió este valor como base ya que es el voltaje de polarización del Microcontrolador. Al desarrollar los sensores, se revisó bajo qué voltajes de polarización se puede utilizar un amplificador operacional y este va en general desde los ± 3.5 Volts hasta los ± 15 Volts, con lo que se decidió armar para este propósito una fuente de alimentación simétrica a ± 5 Volts.

Dada la gran cantidad de dispositivos nuevos que se encuentran hoy en el mercado, se pudo conseguir un relevador con un voltaje de encendido de +5 Volts de corriente directa.

Y es así como se logró estandarizar todo el voltaje del sistema electrónico.

En la Figura A.1 se observa el diagrama de la fuente de alimentación para corriente directa.

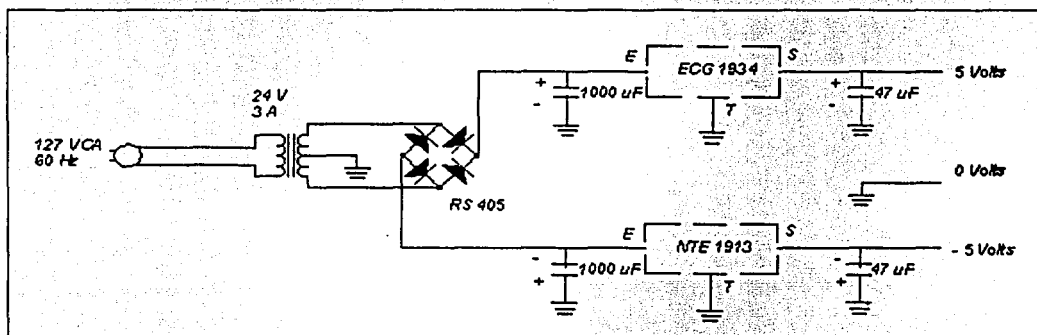


Figura A.1
Diagrama de la fuente de Alimentación.

La fuente básicamente funciona de la siguiente manera, el voltaje recibido de 127 Volts es reducido a ± 12 Volts de corriente alterna todavía.

Posteriormente el voltaje es rectificado con el puente de diodos. Una vez rectificado ambos voltajes, el positivo y el negativo, se filtran a través de un capacitor de 1000 micro farads. Este capacitor reduce el rizo que aún existe después de la rectificación. La etapa siguiente es la de regulación. En ella ambos voltajes pasan por un semiconductor encargado de regular voltaje de corriente directa, el positivo recibe un voltaje de +12 volts y permite una salida de +5 volts. De manera semejante el regulador negativo recibe un voltaje de -12 volts y permite una salida de -5 volts. Después de la etapa de regulación, es conveniente



filtrar una vez más el voltaje de salida de los reguladores, esto se logra con otro par de capacitores, solo que de un valor más pequeño de 47 micro farads.

Nuestro transformador tiene una capacidad de corriente de 3 amperes. Sin embargo, la máxima capacidad de corriente la determina el regulador que tenga menor capacidad de corriente, en este caso es el negativo. Qué nos permite máximo entregar 1.5 amperes.

La fuente de alimentación tiene contemplado un crecimiento de las cargas a conectar a futuro en el prototipo, por ello de los 1.5 amperes disponibles totales, solo se utilizan 375 miliamperes máximo en esta etapa.

A continuación se presenta el desglose de los consumos de los dispositivos instalados.

| Circuito: | Consumo aproximado de corriente: |
|--------------------------------------|---|
| Microcontrolador | 100 mA |
| Sensores(X 3) | $50 \text{ mA} \times 3 = 150 \text{ mA}$ |
| Etapa de potencia | 125 mA |
| TOTAL DE CORRIENTE CONSUMIDA: | 375 mA |



BIBLIOGRAFÍA.

- Sensores y acondicionadores de señal.
Ramón Pallas Areny.
- Ingeniería de software.
Ian Sommerville
Adisson - Wesley Iberoamericana.
- Diseño Digital.
Morris M. Mano
Prentice Hall.
- Introducción a los sistemas de bases de datos.
Date.
- Control de calidad
Dale H. Besterfield.
Prentice Hall.
- Tutorial MICRO en Disquetes.
Tesis de: Germán Figueroa Espinoza.
Facultad de Ingeniería.
UNAM.