



# UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO

---

---

FACULTAD DE INGENIERÍA

## Diseño de un Perfilómetro para Modelos de Fondo Móvil

Eduardo Sámano Aviña  
Ingeniería Eléctrica Electrónica

Director: Ing. Rodolfo Peters Lammel

2008





## Agradecimientos

Esta tesis está dedicada a mis padres, que aunque el día de hoy ya no están conmigo siempre me brindaron su cariño, apoyo, guía y enseñanzas. Regalos sin los cuales no me hubiera sido posible llegar a esta etapa de mi vida.

El esforzarme todos los días para alcanzar mis metas, el perseguir mis sueños a pesar de algunas derrotas, el valorar el estudio como una herramienta para el desarrollo personal son solo algunos de los principios que ellos me inculcaron.

Gracias por ello y por todo.

Aprovecho también para agradecer a la familia Alvarado Castañeda por brindarme su apoyo incondicional. Por haberme hecho parte de ellos.

Gracias familia.

A mis hermanos: Dolores, Héctor, Alberto y Mariana; así como a mis amigos que han formado parte importante de mi vida y con los que he compartido lo mejor de ella.

A mi director de tesis, el Ing. Rodolfo Peters y a los sinodales: M.I. Antonio Salvá, M.I. Juan Fernando Solórzano, M.I. Norma Elva Chávez y el Ing. Rafael Flores por su interés y disposición para lograr este trabajo.

# Contenido

<b>Introducción.....</b>	<b>i</b>
<b>I. Descripción General.....</b>	<b>1</b>
<b>II. Sistema mecánico.....</b>	<b>5</b>
1. Sistema de Posicionamiento del Sensor.....	5
a) Tornillos sinfin .....	6
b) Poleas fijas y móviles, con cables de acero .....	7
c) Rieles y banda dentada (Propuesta Final) .....	8
2. Sistema de soporte.....	9
3. Sistema Generador de Movimiento.....	10
a) Motores de Corriente Directa .....	10
b) Motores paso a paso (Propuesta Final) .....	11
4. Mecanismo.....	12
a) Cálculos.....	13
<b>III. Sistema de Control .....</b>	<b>17</b>
1. Unidad de Automatización.....	20
2. Unidad de Control.....	21
3. Algoritmos .....	22
a) Unidad de Control Central.....	22
<b>IV. Sistema de Procesamiento de la información .....</b>	<b>29</b>
1. Estructuras de Datos a Manejar.....	29
2. Esquema del Área y de los Puntos a Medir.....	31
3. Despliegue de los Valores Medidos (forma tabular).....	33
4. Despliegue de los valores medidos (forma gráfica) .....	35

a) Líneas 2D.....	35
b) Líneas 3D (cintas).....	36
c) Áreas 3D.....	38
5. Impresión de los datos.....	39
6. Impresión de los Gráficos .....	40
7. Almacenamiento de la Información .....	41
a) Archivo de Texto (*.TXT).....	41
b) Archivo de Valores Separados por Comas (*.CSV).....	42
8. Validación de los Datos .....	42
9. Recuperación de Datos Almacenados .....	43
<b>V. Sistema de Comunicación .....</b>	<b>45</b>
1. Unidad de Automatización $\leftarrow \rightarrow$ Unidad de Control Central.....	45
2. Codificación de los Datos .....	49
a) Coordenadas de los Puntos .....	49
b) Resultado de la Medición .....	50
c) Comandos de Petición, Informes de Estado y Errores.....	51
<b>VI. Programa .....</b>	<b>53</b>
1. Modos de Operación y Parámetros Requeridos .....	53
a) Automático .....	54
b) Manual.....	56
<b>VII.Sensor .....</b>	<b>59</b>
1. Sensores Estereoscópicos.....	60
2. Sensores Ultrasónicos .....	62
3. Sensores Láser .....	63
a) Por Triangulación .....	64

b) Por Tiempo de Vuelo.....	65
c) Por Corrimiento de Fase .....	65
4. Conclusiones .....	66
5. Sensor Propuesto.....	68
<b>VIII. Diseño Eléctrico-Electrónico .....</b>	<b>69</b>
1. Fuentes de Alimentación.....	69
2. Sensores de Inicio y Fin de Carrera.....	72
3. Enlace PC $\leftarrow \rightarrow$ Unidad de Control.....	73
4. Unidad de Control.....	74
5. Controladores para los Motores de Pasos.....	76
<b>IX. Mejoras Opcionales.....</b>	<b>79</b>
1. Visión del modelo desde la perspectiva del sensor .....	79
2. Panel manual autónomo .....	80
3. Modo de operación “Experto”.....	81
4. Representación tridimensional del modelo .....	82
5. Graficación de “Curvas de Nivel” .....	83
<b>X. Anexo.....</b>	<b>85</b>





## INTRODUCCIÓN

---

Con la finalidad de automatizar y mejorar las mediciones realizadas en un modelo de fondo móvil, la Comisión Federal de Electricidad se acercó al Instituto de Ingeniería de la UNAM solicitando el diseño y la construcción de un “*Perfilómetro para modelos de fondo Móvil*”.

En este modelo la CFE realiza pruebas de erosión y deslave del fondo del mismo debido al flujo del agua. Para ello inundan un contenedor con agua, que cuenta con un fondo móvil; en nuestro caso arena, y posteriormente abren unas compuertas para permitir el drenado del modelo.

Para conocer la erosión o deslave existente es necesario conocer como se encuentra distribuida la arena después de dicha prueba; para ello se toma la altura de la arena en diferentes puntos.

Actualmente las lecturas se realizan manualmente mediante el uso de una regla y un método de cuadrícula basado en el uso de estambre. Esto tiene como inconveniente la necesidad de que al menos una persona se introduzca en el modelo, distorsionándolo al momento de pisar en él, además de que la medición se vuelve de carácter invasivo al tener que tocar físicamente con la regla a la arena; alterando la medición al sumir con ello la parte superficial de la arena, e incluso generando deslaves.

Al solicitar el apoyo del Instituto de Ingeniería de la UNAM la CFE tiene en mente el salvar estos inconvenientes con un equipo que realice mediciones del tipo no invasivo, con mayor precisión y en menor tiempo, además de realizarlas de forma automática.

## I. DESCRIPCIÓN GENERAL

---

Con este equipo denominado “*Perfilómetro para modelos de fondo Móvil*” se pretende determinar la altura a la que se encuentra la arena del modelo con respecto a un nivel prefijado para que posteriormente se pueda generar una representación del mismo modelo después de haberse efectuado alguna prueba en el.

Dentro de los requerimientos necesarios podemos destacar que el equipo de medición bajo ninguna causa debe de alterar, modificar u obstaculizar el funcionamiento del modelo, así como poder brindar una lectura de la altura con una precisión de +/- 5 mm.

El modelo de fondo móvil consiste en un contenedor de forma rectangular cuyo fondo es cubierto con arena.

## I. Descripción General

---

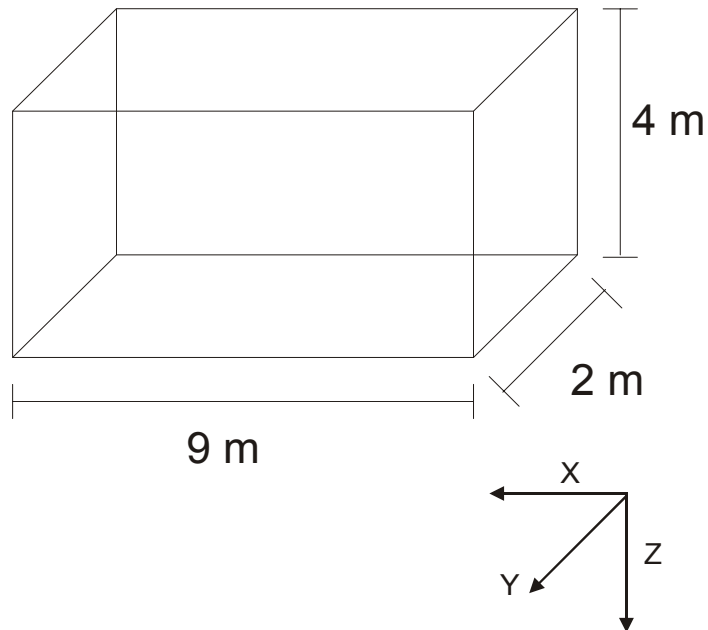


Fig. 1 .- Esquema del modelo.

Con la finalidad de facilitar la comprensión del equipo de medición a diseñar se optó por dividirlo en diferentes sistemas, de acuerdo a su naturaleza y exigencias:

- **Sistema Mecánico**

Refiriéndome con sistema mecánico al encargado de mover, sujetar y posicionar a la cabeza sensora para realizar la medición correspondiente.

- **Sistema de Control**

El encargado de determinar la situación bajo la que opera el sistema para evitar condiciones de error o de mal

funcionamiento, asegurando que el equipo opere de manera apropiada.

- **Sistema de Interacción con el Usuario**

Este sistema está encargado de facilitar la interacción entre el usuario del equipo y el propio equipo, realizando con cierto grado de autonomía todas las mediciones requeridas de acuerdo a las especificaciones dadas por el usuario.

- **Sistema de Medición**

Es la base del equipo, pues es con este sistema con el que se efectúa la medición de la variable bajo estudio.

- **Sistema de Procesamiento de la Información**

Mediante este sistema se pretende procesar la información suministrada por el usuario para determinar los puntos del modelo a examinar, así como procesar y mostrar los resultados obtenidos en las mediciones. Además de mostrar la situación bajo la que opera el sistema (Ocupado, error, etc.)

- **Sistema de Comunicación**

Mediante este sistema las diferentes unidades de control que constituyen el equipo se interrelacionarán, logrando así que trabajen para que en conjunto logren el fin deseado.



## II. SISTEMA MECÁNICO

---

El sistema mecánico a su vez se subdividió en:

- Sistema de posicionamiento del sensor.
- Sistema de soporte.
- Sistema generador de movimiento.
- Mecanismo.

### 1. Sistema de Posicionamiento del Sensor.

Para trasladar al sensor hasta el punto a analizar existieron principalmente 3 propuestas que fueron:

- Mediante el uso de tornillos sinfín.

## **II. Sistema Mecánico**

---

- Mediante el uso de poleas fijas y móviles, con cables de acero.
- Mediante el uso de rieles y de banda dentada (Propuesta Final).

### **a) Tornillos sinfín**

Este método consistía en colocar un tornillo sinfín en ambos lados de los ejes X y Y.

Con estos tornillos sin fin se intentaba alcanzar una buena precisión en el desplazamiento respectivo, gracias a que la cuerda de los tornillos impediría que se desplazara involuntariamente el sistema de sujeción hacia ambos lados de dicho eje. Además de que el control de posición dependería del control del giro de dicho tornillo; lo cual es posible de realizar con muy buena precisión con el uso de sistemas reductores.

Otra ventaja de este método, es que uno de los motores encargados del giro de dichos tornillos tendría que estar fijo en uno de los extremos, con lo que se reduce el peso del sistema móvil; cabe mencionar que dicho motor sería el de mayor tamaño, pues tendría que ser el encargado de mover a las estructuras de los otros dos ejes.

Sin embargo nos enfrentamos a la problemática de encontrar tornillos sin fin de las dimensiones apropiadas a un costo razonable, y en caso de encontrarlos garantizar que los mismos no se fueran a pandear debido a su propio peso y del peso de las estructuras de los ejes que tendrían que



soportar. Lo cual no era fácil de conseguir, debido a la longitud requerida, sobre todo para los del eje X, pues tendrían que tener una longitud de 9 m.

### **b) Poleas fijas y móviles, con cables de acero**

Este método surgió después de analizar el mecanismo utilizado en un graficador profesional, el cual consistía en una serie de poleas fijas y móviles acopladas mediante el uso de cable de acero. Estas últimas se movían al soltar o retraer el cable al ser enrollado en cilindros que se hacían girar mediante motores.

Mientras el cable se soltaba por un lado del cilindro, se iba enrollando por el otro lado. Manteniendo así siempre una longitud constante en el cable suelto. Al estar el sistema móvil sujeto al cable, al desplazarse este, el sistema móvil se movía con él.

Dentro de las ventajas de este sistema, las que nos motivaron a analizarlo, era el bajo costo del cable y de las poleas; así como el permitir que los motores se quedaran fijos en un extremo y así evitar el tener que lidiar con el peso de los mismos.

A pesar de estas ventajas fue imposible utilizar este método, pues para ello era necesario localizar las poleas fijas en diferentes puntos del

## **II. Sistema Mecánico**

---

trayecto al igual que las guías necesarias para garantizar el trayecto de las poleas móviles. Y debido a que como soporte de las mismas solo se cuenta con las paredes del modelo, esto se volvería complejo y poco funcional. Además de que el cable tendría que pasar en varias ocasiones a lo ancho y largo del modelo.

Por ello se optó por buscar otro método.

### **c) Rieles y banda dentada (Propuesta Final)**

Por último analizamos este método el cual satisfacía nuestras necesidades de funcionalidad y de costo.

Este método consiste en colocar a ambos lados del eje X rieles en forma de ángulo, con la cara vertical fija a la pared y en la otra adherir la banda dentada. El uso de la banda dentada es necesario para evitar deslizamientos debido a la inercia del sistema móvil y así poder garantizar con cierta tolerancia la posición de la cabeza sensora. Para evitar la acumulación de suciedad en la banda dentada se propone que sea colocada por la parte inferior de la cara horizontal del ángulo, no siendo crucial para el buen funcionamiento del equipo.

Para afianzar el carro (eje Y, eje Z y sensor) al eje X se emplea un engrane, que será el encargado de hacer mover al carro a lo largo del eje X, y dos poleas por el otro lado (parte superior).

Para el movimiento sobre el eje Y se utiliza el mismo método con la única diferencia que en vez de desplazarse sobre un riel afianzado en la pared, se desplaza sobre el eje móvil que se desplaza sobre el eje X.

En el esquema siguiente se puede ver de forma general como quedaría dicho sistema.

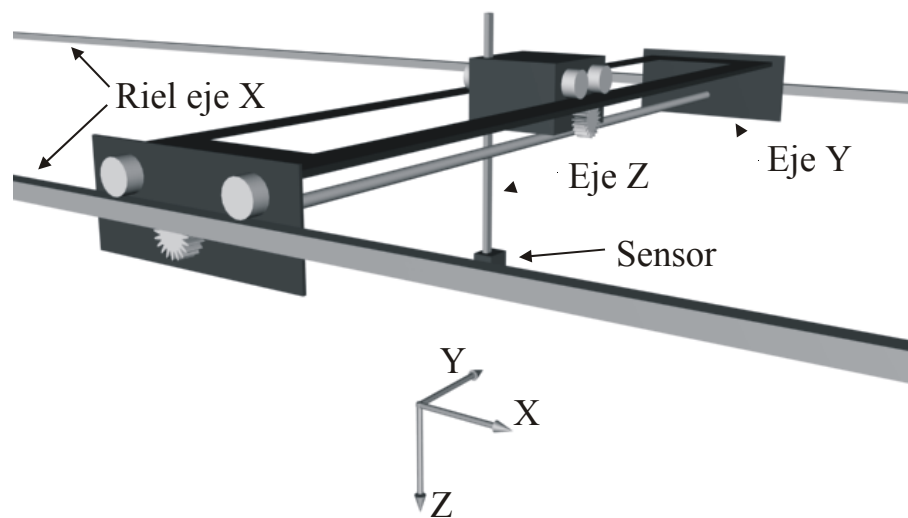


Fig. 2 .-Esquema simplificado del prototipo propuesto

### 2. Sistema de soporte.

## **II. Sistema Mecánico**

---

Para la construcción del equipo se recomienda estructuras huecas de aluminio, con la finalidad de brindar el soporte necesario y a la vez la mayor ligereza posible. Con lo que se conseguiría reducir el tamaño de los motores, con ello el peso y a su vez el costo y la potencia requerida para su operación.

### **3. Sistema Generador de Movimiento.**

Para lograr mover el sensor a través del área bajo estudio se decidió utilizar motores eléctricos, debido a su costo y de la facilidad para controlarlos mediante el uso de la electrónica. Quedando a discusión el tipo de los mismos, considerando:

- a) Motores de Corriente Directa
- b) Motores paso a paso

#### **a) Motores de Corriente Directa**

En un principio se propuso el uso de motores de corriente directa, pues requieren de menor control para establecer el sentido de giro y generar este, además de ser de menor costo que los motores de pasos de la misma

capacidad. Sin embargo al hacer pruebas sobre como controlarlos se tuvieron problemas para frenarlos en el momento preciso, pues por su inercia seguían girando algunas vueltas después de lo deseado. Este problema nos impedía controlar con precisión el punto en el que el sensor sería posicionado para efectuar la lectura correspondiente.

Debido a ello se optó por el uso de motores paso a paso.

### **b) Motores paso a paso (Propuesta Final)**

El motor paso a paso es un dispositivo utilizado para convertir impulsos eléctricos en discretos movimientos mecánicos de rotación. Este tipo de motores requieren una cierta secuencia en la alimentación de sus bobinas para generar dicho movimiento, existiendo una secuencia para el movimiento hacia un lado, y otra para hacerlo en sentido contrario.

Este tipo de motores nos permite controlar con la precisión requerida en cuanto a la velocidad de movimiento y en la posición deseada. Todo esto con cierta facilidad, apoyados en los circuitos integrados existentes hoy en día para ello, de los cuales me ocuparé al explicar el sistema de control del equipo bajo diseño.

## II. Sistema Mecánico

---

### 4. Mecanismo

El mecanismo para desplazar el carro sobre los ejes será el siguiente:

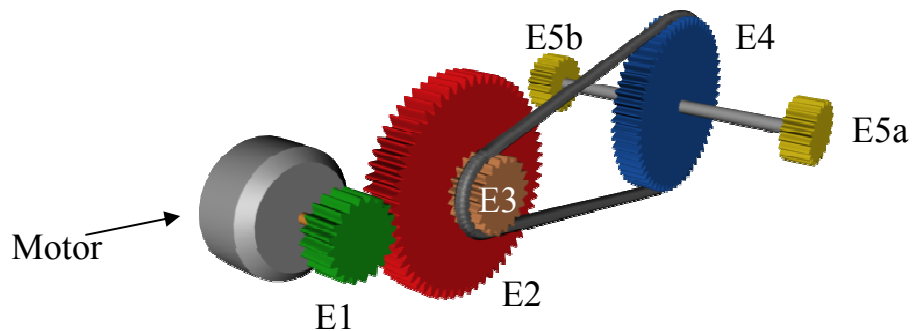


Fig. 3 .- Mecanismo de los Ejes

Cada eje contará con un mecanismo como el anterior.

El parámetro fundamental de diseño de este mecanismo es el de garantizar una precisión de 1mm para las mediciones en el eje correspondiente.

Este sistema es del tipo reductor con la finalidad de mejorar la precisión proporcionada por el motor de pasos e incrementar la fuerza proporcionada por el mismo motor.

Lo anterior con la finalidad de utilizar un motor más pequeño y de pocos pasos; obteniendo un sistema más ligero, económico y eficiente.

Dimensiones sugeridas:

Engrane	D (cm)	Reducción Angular
E1	2	3
E2	6	
E3	2	3
E4	6	
E5, a y b	2	9
<b>Factor de Reducción</b>		

### a) Cálculos

Nomenclatura empleada:

$D_x$  – diámetro de  $E_x$

$l_x$  – desplazamiento lineal de  $E_x$

$da_x$  – desplazamiento angular de  $E_x$  (radianes)

#  $p$  – No. de pasos del motor

Definiciones y relaciones:

## II. Sistema Mecánico

---

$$\ell_5 < 1/2mm \text{ (por parámetro de diseño)} \cdots (I)$$

$$\ell_5 = da_5 \cdot D_5 \cdots (II)$$

$$da_5 = da_4 \text{ (por ser coaxiales)} \cdots (III)$$

$$da_3 = da_2 \text{ (por ser coaxiales)} \cdots (IV)$$

$$da_1 = \frac{2\pi}{\#p} \cdots (V)$$

$$da_4 = \frac{1}{3} da_3 \text{ (por diseño)} \cdots (VI)$$

$$da_2 = \frac{1}{3} da_1 \text{ (por diseño)} \cdots (VII)$$

Sustituyendo V en VII:

$$da_2 = \frac{1}{3} \cdot \frac{2\pi}{\#p} \cdots (VIII)$$

De VIII y IV tenemos:

$$da_3 = \frac{1}{3} \cdot \frac{2\pi}{\#p} \cdots (IX)$$

Sustituyendo IX en VI tenemos:

$$da_4 = \frac{1}{3} \cdot \frac{1}{3} \cdot \frac{2\pi}{\#p} = \frac{2}{9} \cdot \frac{\pi}{\#p} \cdots (X)$$

De X y III tenemos:

$$da_5 = \frac{2}{9} \cdot \frac{\pi}{\#p} \cdots (XI)$$



De III en II tenemos:

$$\ell a_s = \left( \frac{2}{9} \cdot \frac{\pi}{\#p} \right) \cdot D_s \dots (XII)$$

Así, si suponemos un motor con #P=48 tendremos:

$$\ell a_s = \left( \frac{2}{9} \cdot \frac{\pi}{48} \right) \cdot 20 = 0.290 \text{ mm}$$

Con lo que se cumple el enunciado I.

Si despreciáramos el efecto de backlash y la imperfección en la fabricación de los engranes, el error en el desplazamiento de los ejes sería de  $\pm 0.29 \text{ mm}$ .

El error debido al backlash y a la imperfección de los componentes del sistema se podría considerar como un error constante, por lo se podría medir y calibrar el sistema para prácticamente anular su efecto. Cabe aclarar que conforme se vaya desgastando el mecanismo debido a la fricción entre sus componentes dichos errores irían aumentando, por lo que se tendría que determinar dicho error periódicamente y recalibrar el equipo.



### III. SISTEMA DE CONTROL

---

El Sistema de Control será el encargado de procesar la información proporcionada por el usuario y con ello realizar las acciones pertinentes para que el sistema genere resultados satisfactorios, asegurando en cada instante el correcto funcionamiento de todo el sistema.

El Sistema de Control tendrá encomendadas las siguientes tareas:

- Traducir los parámetros proporcionados por el usuario en solicitudes de medición.
- Generar las señales apropiadas para permitir la intercomunicación entre los bloques.

### **III. Sistema de Control**

---

- Revisar periódicamente las señales de los sensores para garantizar la seguridad del usuario y del equipo.
- Verificar que la lectura obtenida en el proceso de medición sea válida.
- Realizar un proceso de logística que contribuya a un mejor desempeño del equipo reduciendo el tiempo de medición
- Proporcionar al usuario la capacidad de intervenir en el proceso cuando este así lo considere necesario (detener, reanudar, reiniciar, etc.)

### III. Sistema de Control

Para realizar todas estas funciones se ha optado por implementar un Sistema de Control distribuido, el cual consta de dos unidades independientes de procesamiento:

- Unidad de Automatización (Principalmente una PC)
- Unidad de Control

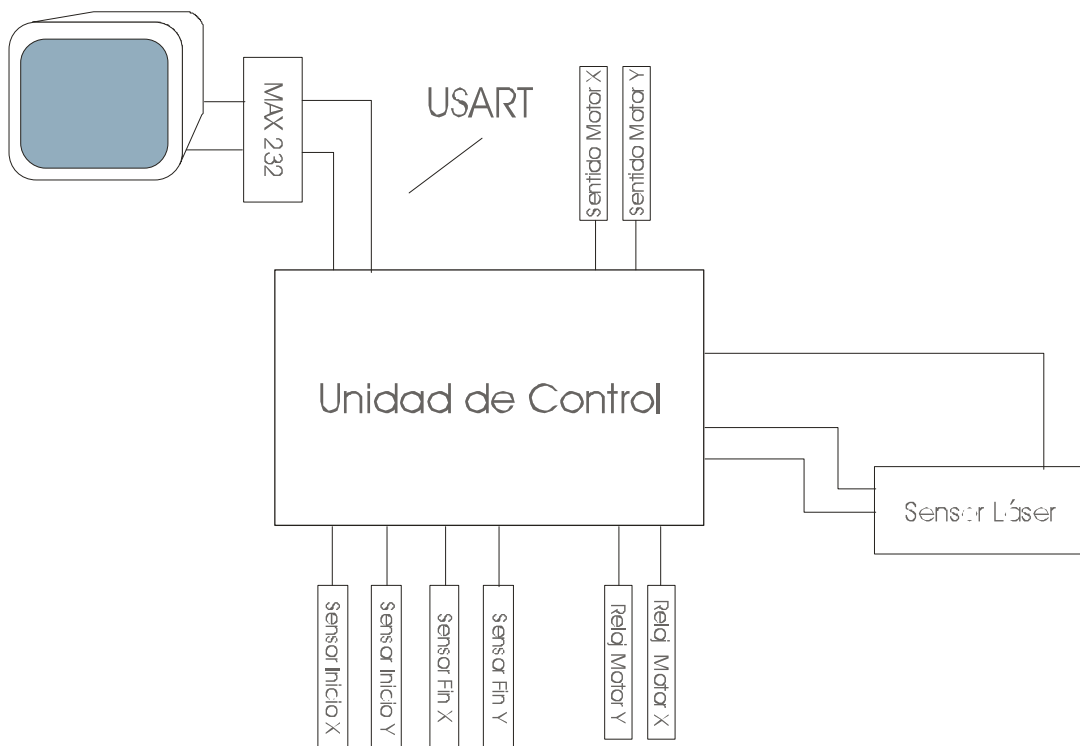


Fig. 4 .- Sistema de Control

### **III. Sistema de Control**

---

#### **1. Unidad de Automatización**

Esta unidad está principalmente constituida por una computadora personal y su programa asociado. Aunque cabe aclarar que la función del programa no es únicamente la de intervenir en el proceso de control, pues también toma parte en otros procesos (procesamiento de la información principalmente)

A esta unidad se le ha dado el nombre de Unidad de Automatización porque será la encargada de que a partir de los parámetros proporcionados por el usuario calcular los puntos a medir (sus coordenadas) e ir transmitiendo de uno en uno dichos puntos hacia la Unidad de Control Central para que sea esta la que realice cada medición.

Con la finalidad de conocer las necesidades del usuario, esta unidad proporciona de forma gráfica (en el monitor de la PC) controles capaces de ser manipulados por el usuario.

Mediante el programa residente en la PC el usuario podrá:

- Conectar / Desconectar a esta unidad del resto del sistema.
- Proporcionar los parámetros que definan los puntos del modelo a medir.

- Posicionar el carro sobre algún punto específico del modelo.
- Cancelar, detener o reanudar el proceso de medición.

## 2. Unidad de Control

Esta unidad que estará principalmente constituida por un microcontrolador será la encargada de generar las señales adecuadas para excitar los controladores de los motores y posicionar así al carro sobre el punto de interés con ayuda de sensores de posición, así como interactuar con el sensor de medición y realizar la propia medición.

Las funciones de esta unidad serán:

- Posicionar en un principio al carro en la esquina de origen, para calcular los desplazamientos futuros a partir de ese punto.
- Calcular el desplazamiento relativo necesario para que partiendo del punto en que se encuentre llegue a posicionarse sobre el siguiente punto solicitado por el usuario.
- Generar las señales necesarias para lograr dicho desplazamiento.

### **III. Sistema de Control**

---

- Comandar al sensor de medición.
- Adquirir el resultado de la medición.
- Transmitir el valor de la lectura realizada hacia la Unidad de Automatización.
- Atender las peticiones de detenerse, reanudar o abortar el proceso de medición en curso.

### **3. Algoritmos**

En este apartado se explicará la secuencia de operaciones que la Unidad de Control Central realizará para cumplir con sus funciones. La Unidad de Automatización no se explica aquí ya que se hará en el capítulo dedicado al programa.

#### **a) Unidad de Control Central**

Algoritmo General

1. Configurar los puertos de comunicación serie.
2. Configurar los puertos de entrada/salida.



3. Monitorear el inicio de comunicación con la PC.
4. Posicionar el carro en la esquina de inicio.
5. Esperar a que se establezca la comunicación con la PC.
6. Informar a la PC que se encuentra lista para responder a peticiones.
7. Esperar datos y/o instrucciones.
8. Ejecutar la instrucción.
9. Responder indicando la existencia de un error, el resultado o que se encuentra lista para recibir una nueva petición.
10. Repetir desde el punto 7.

Algoritmo Posicionamiento en el Origen (igual para los 3 ejes)

1. Establecer sentido de giro del motor hacia el origen (sentido decreciente de acuerdo al eje).
2. Revisar sensor de origen.
3. Si se ha activado el sensor saltar al paso 6.
4. Moverse un paso
5. Repetir desde el paso 2.
6. Fin.

### **III. Sistema de Control**

---

#### Algoritmo Posicionamiento

1. Posicionar Eje X.
2. Posicionar Eje Y
3. Fin

#### Algoritmo Posicionamiento del Eje X

1. Comparar el byte más significativo de la posición destino con el de la posición actual.
2. Si el de la posición destino es mayor al de la posición actual establecer el sentido del giro del motor en sentido creciente, de lo contrario en sentido decreciente. Si son iguales pasar al punto 5.
3. Desplazar el carro en un paso.
4. Repetir desde el punto 1.
5. Comparar el segundo byte de la posición destino con el de la posición actual.
6. Si el de la posición destino es mayor al de la posición actual establecer el sentido del giro del motor en sentido creciente, de lo contrario en sentido decreciente. Si son iguales pasar al punto 9.
7. Desplazar el carro en un paso.

8. Repetir desde el punto 5.
9. Comparar el tercer byte de la posición destino con el de la posición actual.
10. Si el de la posición destino es mayor al de la posición actual establecer el sentido del giro del motor en sentido creciente, de lo contrario en sentido decreciente. Si son iguales pasar al punto 13.
11. Desplazar el carro en un paso.
12. Repetir desde el punto 9.
13. Comparar el cuarto byte de la posición destino con el de la posición actual.
14. Si el de la posición destino es mayor al de la posición actual establecer el sentido del giro del motor en sentido creciente, de lo contrario en sentido decreciente. Si son iguales pasar al punto 17.
15. Desplazar el carro en un paso.
16. Repetir desde el punto 13.
17. Comparar el byte menos significativo de la posición destino con el de la posición actual.
18. Si el de la posición destino es mayor al de la posición actual establecer el sentido del giro del motor en sentido creciente, de lo contrario en sentido decreciente. Si son iguales pasar al punto 21
19. Desplazar el carro en un paso.
20. Repetir desde el punto 17.

### **III. Sistema de Control**

---

21. Fin

#### Algoritmo Posicionamiento del Eje Y

1. Comparar el byte más significativo de la posición destino con el de la posición actual.
2. Si el de la posición destino es mayor al de la posición actual establecer el sentido del giro del motor en sentido creciente, de lo contrario en sentido decreciente. Si son iguales pasar al punto 5.
3. Desplazar el carro en un paso.
4. Repetir desde el punto 1.
5. Comparar el segundo byte de la posición destino con el de la posición actual.
6. Si el de la posición destino es mayor al de la posición actual establecer el sentido del giro del motor en sentido creciente, de lo contrario en sentido decreciente. Si son iguales pasar al punto 9.
7. Desplazar el carro en un paso.
8. Repetir desde el punto 5.
9. Comparar el tercer byte de la posición destino con el de la posición actual.
10. Si el de la posición destino es mayor al de la posición actual establecer el sentido del giro del motor en sentido creciente, de

lo contrario en sentido decreciente. Si son iguales pasar al punto 13.

11. Desplazar el carro en un paso.
12. Repetir desde el punto 9.
13. Comparar el byte menos significativo de la posición destino con el de la posición actual.
14. Si el de la posición destino es mayor al de la posición actual establecer el sentido del giro del motor en sentido creciente, de lo contrario en sentido decreciente. Si son iguales pasar al punto 17.
15. Desplazar el carro en un paso.
16. Repetir desde el punto 13.
17. Fin

#### Algoritmo de Medición

1. Posicionar Carro
2. Toma de Lectura
3. Fin

### **III. Sistema de Control**

---

#### Algoritmo Toma de Lectura

1. Esperar cierto tiempo después del posicionamiento para que se estabilice el sistema.
2. Activar el sensor laser.
3. Solicitar al convertidor analógico digital que realice una conversión.
4. Esperar el tiempo necesario para que se realice la conversión.
5. Leer el dato proveniente del ADC
6. Fin

## **IV. SISTEMA DE PROCESAMIENTO DE LA INFORMACIÓN**

---

En este capítulo se hará una breve explicación de cómo se procesa la información resultante durante un proceso de medición, esto es, estructuras de datos, como se visualizan, se grafican, se imprimen y/o se graban en disco los datos generados.

El procesamiento de la información es realizada principalmente a nivel del programa residente en la PC.

### **1. Estructuras de Datos a Manejar**

Dentro del programa (a nivel de programación) existen definidas las siguientes estructuras.

#### **IV. Sistema de Procesamiento de la Información**

CoorX.- Que representa a un arreglo unidimensional que contiene todos los valores de las coordenadas X de los puntos medidos o a medir. En el modo de operación *automático* (que se explicará en el capítulo “Programa”) este arreglo se irá llenando de información durante el proceso de generar los puntos a medir, y contendrá todos los valores posibles de coordenadas X de forma ascendente. En el modo de operación *manual* este arreglo contendrá las coordenadas X de los puntos medidos en el mismo orden en el que se realicen las mediciones.

CoorY.- Que representa a un arreglo unidimensional que contiene todos los valores de las coordenadas Y de los puntos medidos o a medir. Al igual que CoorX la forma en que se irán generando los valores a almacenar en este arreglo dependerá del modo de operación, siendo similar al de CoorX.

Lecturas.- Que representa un arreglo bidimensional con los valores devueltos como resultado de cada medición. En el modo de operación *automático* cada columna corresponde a un valor de CoorX y cada renglón a un valor de CoorY. Cada celda de este arreglo representa a un punto medido del modelo. En el modo *manual* solo se ocupa una columna en donde cada renglón representará a un punto medido y se irá llenando de forma secuencial, esto es, el valor del primer punto medido irá en el primer renglón, el del segundo en el segundo renglón y así



## **IV. Sistema de Procesamiento de la Información**

---

sucesivamente. Dado el origen de los datos que se emplean para formar este arreglo, este arreglo se va llenando después de realizar una medición.

### **2. Esquema del Área y de los Puntos a Medir**

Ya que el proceso de medición puede ser algo tardado, se ha considerado pertinente generar un esquema que represente al modelo y a los puntos a medir antes de realizar cualquier medición. Esto con la finalidad de que el usuario pueda comprobar si los parámetros que introdujo son los adecuados para realizar la medición que requiere.

Este esquema da la impresión de estar viendo al modelo desde arriba, por lo que la representación del modelo será un rectángulo de 9000 x 2000 unidades<sup>1</sup>, lo que equivale al modelo rectangular de 9 x 2 m. Este rectángulo será de color rojo con una marca en la esquina superior izquierda que representa la esquina de origen.

---

<sup>1</sup> El área disponible para ser medida puede variar dependiendo de los requerimientos físicos del sistema de posicionamiento del sensor.

#### IV. Sistema de Procesamiento de la Información

---

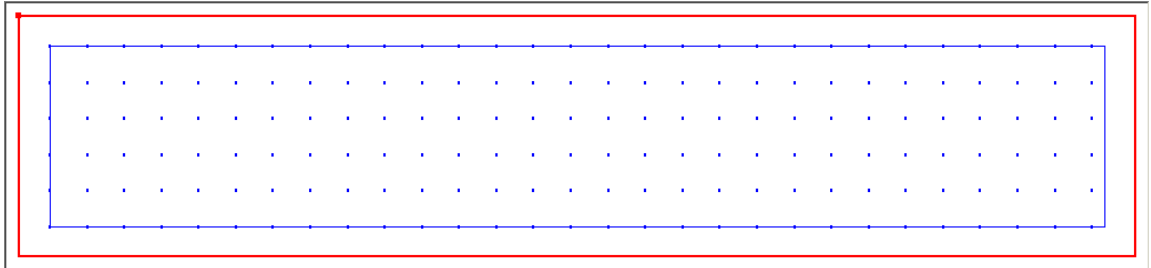


Fig. 5 .- Esquema del Área y de los Puntos a Medir

En esta representación también habrá un rectángulo azul que representara los límites del área a medir, de acuerdo a los parámetros introducidos por el usuario. Este rectángulo tiene la finalidad de mostrar el área que será considerada para calcular los puntos a medir, esto es, todos los puntos a medir estarán localizados dentro o en los límites de dicha región pero nunca afuera.

Por último el esquema tendrá una serie de puntos azules, donde cada uno representa un punto a medir. Si el paso establecido por el usuario entre punto y punto es muy reducido, en vez de ver puntos se podrían ver líneas continuas e incluso áreas totalmente coloreadas, esto debido a las limitaciones de tamaño del esquema en pantalla.

Cabe aclarar que este esquema solo pretende ser una representación aproximada del modelo y de los puntos a medir con la finalidad de orientar al usuario y no una representación precisa.

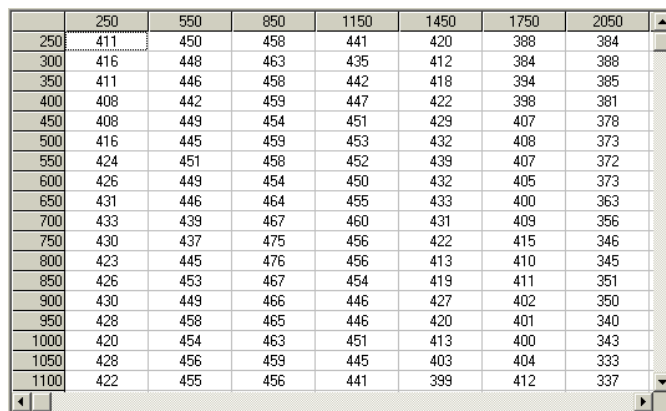
## IV. Sistema de Procesamiento de la Información

En el modo de operación manual solo existirá el rectángulo rojo que representa al modelo y un punto azul, que representará el punto a medir.

### 3. Despliegue de los Valores Medidos (forma tabular)

Durante la ejecución del programa existe una ventana destinada a representar los valores resultantes del proceso de medición, obviamente antes de realizar las mediciones esta ventana se encontrara vacía.

En el modo automático, los valores estarán representados en forma matricial, donde cada celda representará el valor medido en un punto. Cada columna o renglón tendrán un nombre o mejor dicho un valor de coordenada. Las columnas representan a coordenadas del Eje X y los renglones a coordenadas del eje Y.



	250	550	850	1150	1450	1750	2050
250	411	450	458	441	420	388	384
300	416	448	463	435	412	384	388
350	411	446	458	442	418	394	385
400	408	442	459	447	422	398	381
450	408	449	454	451	429	407	378
500	416	445	459	453	432	408	373
550	424	451	458	452	439	407	372
600	426	449	454	450	432	405	373
650	431	446	464	455	433	400	363
700	433	439	467	460	431	409	356
750	430	437	475	456	422	415	346
800	423	445	476	456	413	410	345
850	426	453	467	454	419	411	351
900	430	449	466	446	427	402	350
950	428	458	465	446	420	401	340
1000	420	454	463	451	413	400	343
1050	428	456	459	445	403	404	333
1100	422	455	456	441	399	412	337

Fig. 6 .- Despliegue de los Datos (Modo Automático)

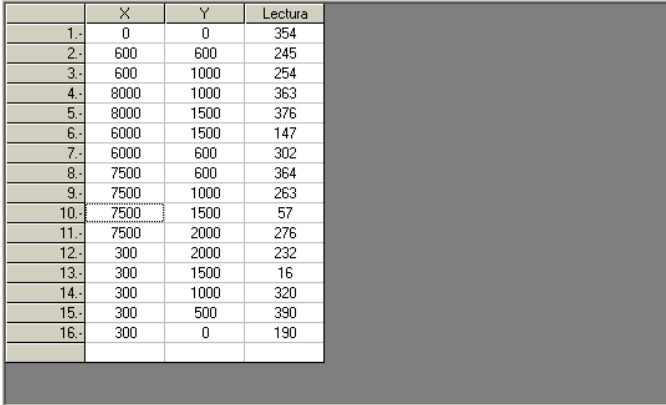
## IV. Sistema de Procesamiento de la Información

---

Por lo que las coordenadas de un punto serán (Columna, Renglón). Los valores de las columnas y de los renglones dependerán de los parámetros introducidos por el usuario.

En el modo manual, los valores estarán representados por una tabla, donde cada renglón contendrá los datos referentes a un punto. Esta tabla estará formada por tres columnas donde la primera se referirá a la coordenada X del punto, la segunda a la coordenada Y, y por último la tercera al resultado de la medición.

Esta tabla irá agregando automáticamente renglones al ir realizando las mediciones.



	X	Y	Lectura
1.-	0	0	354
2.-	600	600	245
3.-	600	1000	254
4.-	8000	1000	363
5.-	8000	1500	376
6.-	6000	1500	147
7.-	6000	600	302
8.-	7500	600	364
9.-	7500	1000	263
10.-	7500	1500	57
11.-	7500	2000	276
12.-	300	2000	232
13.-	300	1500	16
14.-	300	1000	320
15.-	300	500	390
16.-	300	0	190

Fig. 7 .- Despliegue de los Datos (Modo Manual)

El valor de todas las mediciones está dado en milímetros.

### **4. Despliegue de los valores medidos (forma gráfica)**

Si se utilizó el modo de operación automático el usuario tendrá la posibilidad de seleccionar datos contiguos de la representación matricial de las lecturas y solicitar al programa que grafique dichos valores.

De forma predeterminada los datos serán tomados por renglones para generar las series a graficar. Sin embargo el usuario tendrá la libertad de escoger que los datos sean tomados en columnas.

Existen 3 formas diferentes de graficar los datos:

#### **a) Líneas 2D**

En este tipo de gráfica, cada serie será representada por una línea continua, en donde cada vértice representa un punto de los datos seleccionados.

La representación es solo aproximada, pues las líneas se forman al unir los puntos con líneas rectas y generalmente el modelo real no sigue esa trayectoria.

Si se seleccionan datos para generar más de una serie de ellos estas se sobrepondrán en la gráfica.

## IV. Sistema de Procesamiento de la Información

---

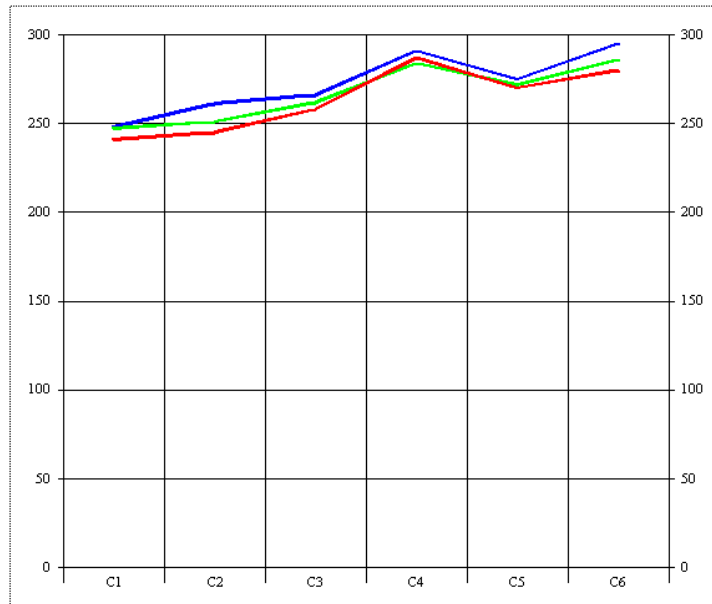


Fig. 8 .- Gráfico Líneas 2D

El usuario podrá escoger si se dibujan marcas en cada punto o no, de forma predeterminada no se dibujarán. Si solo se grafican pocos puntos esto puede ser funcional, pero si se grafican varios puntos las marcas pueden ser una molestia.

### b) Líneas 3D (cintas)

En este tipo de gráfico los datos se graficarán en perspectiva, dibujando una especie de cinta plana por cada serie de datos en vez de líneas simples.

## IV. Sistema de Procesamiento de la Información

Si se seleccionan más de una serie, las cintas se irán colocando una tras otra con efecto de perspectiva.

El usuario tendrá la posibilidad de rotar el gráfico en tres dimensiones con la finalidad de tener el mejor punto de vista de los datos.

Este tipo de representación gráfica tiene la ventaja de que simula de forma burda la superficie del modelo.

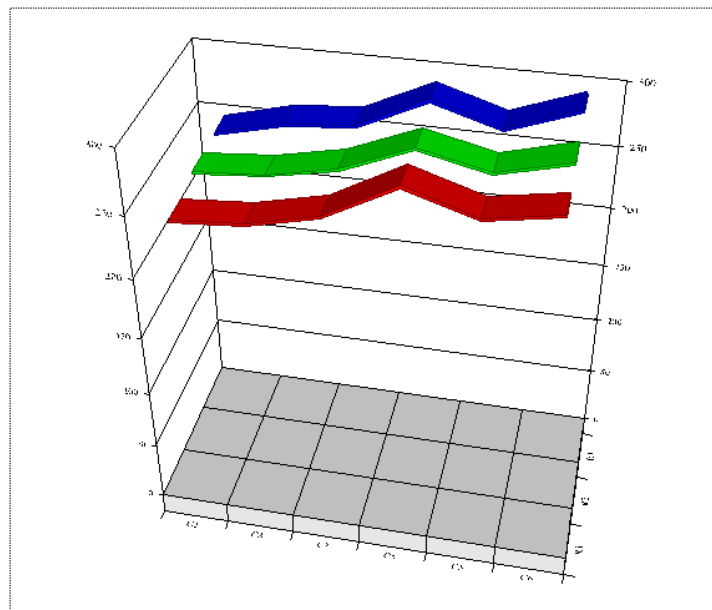


Fig. 9 .- Gráfico Líneas 3D

En este tipo de gráfico el usuario también tendrá la opción de poner marcas en cada punto similar al de las líneas 2D.

## **IV. Sistema de Procesamiento de la Información**

### **c) Áreas 3D**

El gráfico de tipo Áreas 3D es muy similar al de Líneas 3D, con la única diferencia de que desde las cintas se prolonga un volumen hasta la base del gráfico.

La gran ventaja de utilizar este tipo de gráfico con respecto al anterior es que en este tipo se tiene una mejor noción de la perspectiva, pues en el modo anterior, se puede confundir la separación entre series con desplazamiento vertical, cometiendo un error de apreciación. En este tipo de gráfico no se evita el cometer ese error, pero se reduce notablemente.

Al igual que en los tipos anteriores el usuario tendrá la opción de seleccionar si los datos se grafican por renglones o por columnas, así como seleccionar si se dibujan las marcas en cada punto.



## IV. Sistema de Procesamiento de la Información

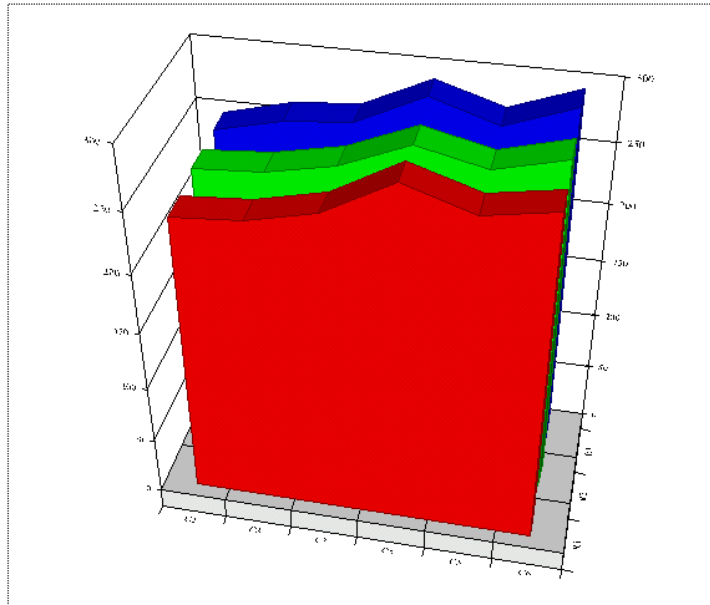


Fig. 10 .- Gráfico Áreas 3D

Si se colorean todas las series con el mismo color y se grafican varias series se puede generar un gráfico muy representativo (más no real) del modelo bajo análisis.

### 5. Impresión de los datos

Existen dos maneras de imprimir los datos, una es la de hacerlo desde el programa y la otra es la de copiar los datos a otra aplicación (principalmente hojas de cálculo) e imprimirlas desde dicha aplicación.

Dado que el programa no pretende ser un programa sofisticado hecho por especialistas que cuentan con varias herramientas durante el diseño, el

## **IV. Sistema de Procesamiento de la Información**

imprimir desde el programa tiene la desventaja de no tener la oportunidad de personalizar el modo de hacerla, solo se tendrá la opción de escoger la impresora destino. Sin embargo esto le permite al usuario imprimir los datos directamente ahorrando tiempo y la necesidad de contar con otra aplicación.

Por otro lado si el usuario tiene acceso a ese tipo de aplicaciones puede ver como una gran utilidad el poder personalizar su impresión cambiando el color de los datos, su tamaño o agregar comentarios entre otras opciones.

### **6. Impresión de los Gráficos**

Al igual que en la impresión de los datos, los gráficos se pueden imprimir directamente o copiarlos a otra aplicación.

El imprimir directamente los gráficos tiene la ventaja de que es más rápido, pero con la desventaja de no poder definir el tamaño ni en que posición de la hoja se realizaría dicha impresión. Además de no poder agregar texto o comentarios.

Sin embargo el usuario podrá pegar el gráfico como imagen (meta archivo) en otra aplicación y así redimensionarla o agregarle texto, incluso pegar el gráfico con los datos fuentes del mismo e imprimirlos juntos.

### **7. Almacenamiento de la Información**

Los datos resultantes del proceso de medición se podrán salvar en disco. La forma en que se almacenarán los datos dependerá del modo de operación y de la preferencia del usuario.

Existirán dos formatos de archivos diferentes:

#### **a) Archivo de Texto (\*.TXT)**

Los datos se almacenarán en formato ASCII dando la posibilidad de poder ver el archivo en cualquier editor de texto.

En modo automático el usuario podrá elegir si los datos se almacenan en forma de lista o en forma matricial. En forma de lista cada renglón tendrá tres valores: Coordenada X, Coordenada Y y Lectura; generando un renglón por cada punto analizado. En forma matricial se grabarán los datos de la misma forma en que se despliegan en el programa.

En el modo manual solo se podrá almacenar los datos en forma de lista.

## **IV. Sistema de Procesamiento de la Información**

### **b) Archivo de Valores Separados por Comas (\*.CSV)**

En este formato los datos también se almacenarán en formato ASCII, con la única diferencia de que en este formato entre valores se insertará una coma. Lo cual permitirá a algunos programas separar de forma automática los valores y generar la tabla correspondiente. Dentro de estos programas están las hojas de cálculo que automáticamente posicionan cada dato en su columna y renglón correspondiente. Permitiendo así que los datos sean abiertos en otras aplicaciones, ya sea para su impresión, graficación o análisis. Incluso hay programas de este tipo que reconocen a este tipo de archivos como propios abriéndolos automáticamente al hacer doble clic en ellos.

La forma de almacenar los datos dependiendo del modo de operación es igual que en el formato anterior.

### **8. Validación de los Datos**

Al inicio del archivo (en la primer línea) se incluirá una cadena de validación en la que estará codificado la forma que se utilizó para generar el archivo (tabular, lista), así como el número de registros que contiene. Esto con la finalidad de poder interpretar el archivo de forma correcta y evitar que modificaciones hechas por el usuario en el archivo vaya a

## **IV. Sistema de Procesamiento de la Información**

generar interpretaciones incorrectas. Si los datos del archivo no coinciden con los datos codificados en la cadena de validación se generará un error y se suspenderá la lectura de los datos.

Esta cadena de validación solo tendrá importancia si se intenta abrir el archivo dentro del visor proporcionado junto con el programa principal, en caso de que se abra el archivo en una aplicación de terceros, esta cadena no tendrá importancia alguna ni afectará la forma en que se lean los archivos.

### **9. Recuperación de Datos Almacenados**

Junto con el programa que controla al sistema desde la PC se distribuye un programa adicional capaz de acceder a los datos almacenados en disco para que el usuario pueda revisar o consultar los datos generados en un proceso de medición antiguo y tener la posibilidad de generar los gráficos pertinentes con dichos datos.

Esto se hace con la finalidad de que el usuario no tenga la necesidad de contar con aplicaciones de terceros para poder revisar sus datos. Sin embargo como no es la finalidad de este proyecto el análisis de los datos obtenidos sino que se limita a la obtención de los mismos, este programa no proporciona herramientas para ello, solo se limita a desplegar los datos y realizar gráficos de los mismos sin que el usuario tenga mucho

#### **IV. Sistema de Procesamiento de la Información**

---

control sobre ello. Las funciones serán las mismas con las que el usuario cuenta en el programa principal.

## V. SISTEMA DE COMUNICACIÓN

---

En este capítulo se explicará la forma en que las unidades de control se comunican para poder interactuar de forma adecuada.

Esto debido a que se emplea un sistema de control distribuido, que consta de dos unidades de control independientes pero interrelacionadas.

### 1. Unidad de Automatización $\leftrightarrow$ Unidad de Control Central

Como se explica en el capítulo dedicado al *Sistema de Control* la unidad de Automatización se encargará de hacer llegar las peticiones del usuario a la unidad de Control Central para que esta unidad las ejecute.

## V. Sistema de Comunicación

---

El enlace entre estas dos unidades se hará mediante el puerto serial de la PC y el puerto USART del microcontrolador de la Unidad de Control Central. Los detalles de la implementación física del enlace se describirán en el capítulo de *Diseño Eléctrico-Electrónico*.

Desde el punto de vista lógico se hará un enlace serial asíncrono con una velocidad de transmisión de 9600 baudios utilizando 8 bits de datos, un bit de parada y sin bit de paridad.

Una vez que el sistema esté energizado en su totalidad y el programa esté siendo ejecutado en la PC el usuario tendrá que indicarle al programa que desea conectarse con el sistema. En este momento el programa abrirá el puerto de comunicación serial elegido por el usuario e intentará enviar un *byte de inicio de comunicación* y esperará a que la Unidad de Control Central le conteste con un *byte de bienvenida*, si esto no sucede en un tiempo predefinido, el programa lo volverá a intentar hasta 3 veces; después de lo cual si no se obtiene una contestación válida el programa informará que no es posible realizar la comunicación. El usuario tendrá la libertad de volver a intentar establecer la comunicación las veces que lo desee.

Debido a esto es importante que el usuario compruebe que el sistema esté energizado antes de solicitar que se establezca la comunicación.



Ya que esté establecida la comunicación la PC esperará a que la Unidad de Control Central le informe que se encuentra lista para recibir peticiones, lo cual hará una vez que haya concluido su proceso de inicialización (retraer en su totalidad la cabeza sensora y posicionar el carro en la esquina de origen).

Después de conocer que la Unidad de Control Central está lista para recibir peticiones el usuario podrá solicitar al programa que realice el proceso de medición.

Para realizar la medición de un punto, el programa enviará las coordenadas de dicho punto y a continuación un *byte de inicio de medición*. Si dicho byte no se envía el sistema no realizará la medición.

La Unidad de Control Central al recibir el *byte de inicio de medición* procederá a posicionar la cabeza sensora sobre dicho punto. Durante este proceso el microcontrolador de esta unidad estará pendiente de las instrucciones enviadas por la PC, con la finalidad de poder responder a las peticiones de pausar, continuar o detener el proceso.

Después de haberse efectuado la medición del punto la unidad de Control Central será la encargada de enviar el resultado de la medición a la PC, y esperará a que el programa le confirme la recepción de dicho valor y a su vez informará que se encuentra lista para recibir una nueva petición. En el caso de no recibir la confirmación volvería a intentar el envío hasta un

## V. Sistema de Comunicación

---

total de 3 veces<sup>2</sup>, y en caso de no recibir contestación el sistema se congelaría requiriendo ser reinicializado.

En el modo de operación automático, el programa enviará la petición de realizar la medición en un punto y esperará a recibir el resultado de dicha medición antes de enviar la siguiente petición.

Si la medición no es posible la Unidad de Control Central enviará el *byte de error de medición* a la PC y el usuario sería el encargado de determinar que procedería (reintentar, continuar con los demás puntos, abortar el proceso).

Si por alguna causa la Unidad de Control Central recibiera la petición de realizar una *medición* durante la ejecución de otra, responderá a dicha petición con un *byte de error por sistema ocupado*.

Si en algún momento los datos recibidos por alguna de estas unidades no cumplen las especificaciones de la codificación de datos entonces contestará con un *Error por Datos Incorrectos* obligando a que sean retransmitidos.

---

<sup>2</sup> Con un intervalo de 5 s. entre intentos.

### 2. Codificación de los Datos

Para que las diferentes unidades intercomunicadas puedan entenderse es necesario establecer como serán enviados e interpretados los datos a transmitir o recibidos. Cabe aclarar que cualquier dato transmitido tendrá el tamaño de un byte, esto es, cualquier dato, comando, error, etc. solo estará conformado por un byte.

La codificación empleada es la siguiente:

#### a) Coordenadas de los Puntos

Las coordenadas de los puntos se enviarán codificadas en código BCD, enviando cada dígito de forma independiente.

Los cuatro bits menos significativos contendrán el valor BCD del dígito correspondiente mientras que los siguientes 4 bits indicarán a que dígito se refieren.

Si alguna coordenada no cuenta con todos sus dígitos, se agregarán ceros a la izquierda hasta completar los cuatro dígitos y siempre se enviarán los nueve dígitos que representan a las coordenadas.

## V. Sistema de Comunicación

---

Parámetro	Código
1 <sup>er</sup> . Dígito Coordenada X	0000XXXX
2 <sup>o</sup> . Dígito Coordenada X	0001XXXX
3 <sup>o</sup> . Dígito Coordenada X	0010XXXX
4 <sup>o</sup> . Dígito Coordenada X	0011XXXX
5 <sup>o</sup> . Dígito Coordenada X	1011XXXX
1 <sup>er</sup> . Dígito Coordenada Y	0100XXXX
2 <sup>o</sup> . Dígito Coordenada Y	0101XXXX
3 <sup>o</sup> . Dígito Coordenada Y	0110XXXX
4 <sup>o</sup> . Dígito Coordenada Y	0111XXXX

Los dígitos de las coordenadas se transmitirán en el mismo orden que en la tabla anterior.

### b) Resultado de la Medición

El resultado de la medición estará siempre constituido por tres cifras, completando con ceros a la izquierda en caso de no ser así.

Este valor al igual que las coordenadas será codificado en código BCD y se transmitirá de acuerdo a la siguiente tabla.

El bit más significativo siempre será 0.

Parámetro	Código
1 <sup>er</sup> . Dígito Lectura	0000XXXX
2 <sup>o</sup> . Dígito Lectura	0001XXXX
3 <sup>o</sup> . Dígito Lectura	0010XXXX

Aunque se utilizará el mismo código que el de las coordenadas no existirá confusión, ya que la interpretación dependerá de la unidad que reciba estos datos. Las coordenadas solo serán transmitidas desde la PC hacia la Unidad de Control Central. Y el resultado de la medición se transmitirá de la Unidad de Control Central hacia la PC.

### c) Comandos de Petición, Informes de Estado y Errores

Todos los Comandos, Informes de Estado y Errores tendrán el bit más significativo igual a 1.

Parámetro	Código
Inicio de Comunicación	10000001
Bienvenida	10000010
Inicio de Medición	10000011
Unidad Lista	10000100
Pausar	10000101
Continuar	10000110

## V. Sistema de Comunicación

---

Parámetro	Código
Detener	10000111
Recepción del valor de la medición (en la PC)	10001000
Unidad Ocupada	10001010
Error por Sistema Ocupado	10001100
Error de Medición	10001110
Error por Datos Incorrectos	11111111

## VI. PROGRAMA

---

En este capítulo se explica el funcionamiento del programa instalado en la *Interfaz con el Usuario*, el cual permitirá que el usuario pueda interactuar con el sistema.

### 1. Modos de Operación y Parámetros Requeridos

Para brindarle al usuario la libertad necesaria para que consiga los resultados que requiere se han considerado dos diferentes formas de operación: uno que le brinde cierta automatización en el proceso y otra que le permita con toda libertad posicionar el *carro* en el punto que considere adecuado y que realice la medición cuando lo desee.

A continuación se describen estos dos modos de operación.

## VI. Programa

---

### a) Automático

En este modo, el usuario tendrá que introducir un total de seis parámetros:

- Límite inferior en el Eje X (IX)
- Límite superior en el Eje X (SX)
- Límite inferior en el Eje Y (IY)
- Límite superior en el Eje Y (SY)
- Paso entre puntos en el sentido del Eje X (PX)
- Paso entre puntos en el sentido del Eje Y (PY)

Una vez que el usuario haya facilitado estos seis parámetros, el programa se encargará de determinar el área rectangular del modelo que será analizada. Esto lo hace a partir de los primeros cuatro parámetros.

Posteriormente el programa generará dos listas, una correspondiente a los valores de coordenadas X (CoorX) y otra de valores de coordenadas Y (CoorY). El procedimiento para calcular las coordenadas X es partir de IX (el primer valor de coordenadas en X) e ir incrementando dicho valor en PX unidades generando en cada iteración un nuevo valor de la lista CoorX hasta alcanzar el mayor valor posible que sea igual o menor a SX. De igual manera se calculan los valores de coordenadas para el Eje Y.



Todas las posibles combinaciones entre estas dos listas serán las coordenadas de los puntos a analizar.

Con la finalidad de brindarle al usuario un apoyo visual, el programa generará un esquema que contenga:

- Un rectángulo que represente los límites del modelo (rojo)
- Un rectángulo que represente el área a analizar (azul)
- Los puntos a analizar (azules)

Si los puntos calculados satisfacen los requerimientos del usuario entonces este podrá ordenar que se comience el proceso de medición. En caso contrario podrá modificar los parámetros anteriores hasta conseguir lo que desea.

En el proceso de medición el programa irá solicitando de una en una las mediciones de cada punto esperando entre cada una el valor resultante de dicho proceso.

Durante el proceso de medición el usuario tendrá la libertad de detener, reanudar o abortar dicha operación.

La medición se realizará en el siguiente orden: empezará en el punto con coordenadas (IX, IY), e irá recorriendo todos los valores de la lista

## **VI. Programa**

---

CoorY manteniendo el valor IX. Una vez realizada la medición de toda esta línea, la coordenada X tomará el siguiente valor de CoorX e irá recorriendo los valores de CoorY en sentido inverso hasta llegar a IY. Y continuará así hasta haber barrido todos los puntos.

Una vez que se hayan hecho todas las lecturas, el programa mostrará los datos obtenidos en forma matricial.

El usuario tiene la posibilidad de seleccionar todos los datos o parte de estos y generar la gráfica que los represente.

Si el usuario así lo desea podrá salvar en disco los datos obtenidos.

### **b) Manual**

Este es el otro modo de operación, con este modo se pretende que la medición dependa únicamente de los parámetros proporcionados por el usuario y no de los cálculos hechos a partir de los mismos.

Los parámetros que el usuario tendrá que introducir en la computadora serán:

- Coordenada X del punto a medir
- Coordenada Y del punto a medir

Una vez que el usuario haya definido estos dos parámetros estará en la posibilidad de solicitar que el sistema realice la medición de ese punto, con lo cual el sistema desplazará el *carro* desde su ubicación actual hasta el punto solicitado y realizará la medición correspondiente.

En un principio se consideró la posibilidad de que en este modo de operación el usuario controlara de forma directa la navegación del *carro*, esto es, que a través del mouse o del teclado el usuario fuera desplazando al *carro* en vez de introducir las coordenadas y dejar que el sistema realizara el desplazamiento. Pero ya que el usuario no tiene acceso visual al modelo hacer esto no genera ninguna ventaja, sin embargo, si se instalara una mini cámara en la cabeza sensora, esta forma de operar el desplazamiento sería muy práctica pues podría ir desplazándose y observando la forma del modelo al mismo tiempo y así escoger los puntos a medir.

En este modo de operación el usuario irá generando una lista con las coordenadas de los puntos medidos y el valor de la medición realizada en cada uno.

Dado que en este modo los puntos pueden no tener nada que ver uno con otro es imposible realizar representación gráfica alguna de ellos.

El usuario tendrá la posibilidad de grabar en disco dicha lista de puntos y valores.



## VII. SENSOR

---

En este capítulo se analizarán los diferentes tipos de sensores capaces de medir distancias existentes en el mercado. Así como su modo de operación y las ventajas y desventajas de su utilización en este proyecto.

Dado los parámetros de diseño requerimos que la medición se realice de manera no invasiva, esto es, que en ningún momento se interactúe mediante el contacto físico con el fondo móvil del modelo.

Debido a esto nos hemos visto limitados a utilizar sensores que empleen algún tipo de onda que nos permita palpar el fondo móvil del modelo sin que esta modifique o altere al mismo.

## VII. Sensor

---

En base a esto se consideraron los siguientes sensores:

- Sensores Estereoscópicos
- Sensores Ultrasónicos
- Sensores Láser

### 1. Sensores Estereoscópicos

Me refiero a sensores estereoscópicos a aquellos que utilizan el mismo fenómeno que el que utilizamos los seres humanos y algunos animales para poder determinar qué tan lejos o cercas se encuentran los objetos de nuestro alrededor, así como el empleado para fines topográficos mediante el uso de fotografías aéreas o satelitales.

De forma muy simplificada el método es el siguiente:

1. Se toman dos fotografías desde dos puntos diferentes con líneas de visión paralelas y con distancias relativas conocidas.
2. Se distingue en ambas fotos la ubicación del punto bajo estudio.
3. Se cuantifica el desplazamiento de dicho punto con respecto a la otra foto.

4. Con las condiciones bajo las que se tomaron las fotos y el desplazamiento aparente del mismo punto en ambas fotos se calcula la altura de dicho punto.

Si el punto a analizar se encuentra más cerca del dispositivo encargado de tomar la fotografía se tendrá un mayor desplazamiento aparente entre ambas fotografías, por el contrario, mientras más retirado esté el desplazamiento aparente es menor.

Este tipo de sensor consta de dos cámaras fijas entre sí a cierta distancia (se puede emplear una sola y desplazarla, del mismo modo que se hace en las fotografías aéreas y en las tomadas por los satélites), y de un procesador de imágenes capaz de detectar automáticamente los puntos a analizar y hacer los cálculos correspondientes para determinar la altura de dichos puntos.

En nuestro caso el procesamiento de las imágenes se vuelve muy complicado e incluso imposible de realizar, esto debido a que poder diferenciar de forma automática el mismo punto de tierra o arena en ambas fotos no es posible, ¿Cómo diferenciar entre dos granos de tierra o arena?

Es por ello que este mecanismo no es útil en nuestro proyecto y se tuvo la necesidad de analizar otro tipo de sensores.

## VII. Sensor

---

### 2. Sensores Ultrasónicos

Este tipo de sensores es actualmente uno de los más utilizados para construir instrumentos portátiles destinados a realizar mediciones de distancia debido a su bajo costo y buen desempeño.

Este tipo de sensor determina la distancia existente entre cierta superficie y la ubicación del sensor.

Su forma de operación es el siguiente:

1. Emiten un pulso ultrasónico hacia la superficie que se desea analizar.
2. Inicializa un contador de tiempo.
3. Recibe el eco del pulso ultrasónico reflejado por la superficie.
4. Detiene el contador.
5. En base al contador y considerando la velocidad de propagación del ultrasonido el sensor calcula la distancia recorrida por la señal ultrasónica desde el emisor hasta la superficie y su regreso. Dividiendo dicha distancia entre dos se obtiene la distancia entre el sensor y la superficie.

A los sensores que utilizan este método de medición se les conoce como sensores de tiempo de vuelo, pues su lectura depende del tiempo que



tarda la señal en “volar” desde el sensor hasta el punto a analizar y regresar.

Este tipo de sensores pueden tener precisión del orden de  $\pm 5\text{mm}$ , dependiendo de su rango de medición.

El inconveniente de los sensores ultrasónicos es que el ultrasonido se propaga de forma cónica, siendo imposible realizar una medición puntual. Parámetro requerido para el diseño del proyecto.

### 3. Sensores Láser

La gran ventaja de este tipo de sensores es que el área de medición puede ser controlada (en el momento de fabricación), pudiendo ser puntual. Esto debido a que el rayo láser que emplean viaja grandes distancias sin abrirse considerablemente.

Estos sensores no requieren que todo el rayo emitido regrese, son sensores tan sensibles que pueden realizar la medición aunque sea escasa la luz reflejada hacia el sensor. Obviamente que la precisión de la medición dependerá de ello, siendo más precisa mientras más reflejante sea la superficie bajo análisis.

## VII. Sensor

---

Dentro de los sensores láser existen tres tipos principalmente, variando en el método utilizado para realizar la medición:

- a) Por Triangulación
- b) Por Tiempo de Vuelo
- c) Por Corrimiento de Fase

### a) Por Triangulación

Este tipo de sensores emiten un rayo láser continuo hacia la superficie bajo análisis y detecta la luz reflejada por la misma, cuantificando qué tanto se desvía el rayo reflejado del rayo emitido. Esto lo hace ya sea por dispositivos CCD<sup>3</sup> o PSD<sup>4</sup>, generando una señal digital o analógica respectivamente.

El ángulo de incidencia del rayo reflejado por la superficie estará determinado por la distancia existente entre el sensor y dicha superficie, por lo que cuantificando dicho ángulo se puede determinar la distancia de la superficie con respecto al sensor.

Este tipo de sensores puede llegar a tener precisión del orden de las micras. Sin embargo mientras mayor sea su precisión menor será su

---

<sup>3</sup> Charge Coupled Device

<sup>4</sup> Position Sensitive Detector

rango de medición. Para los sensores con precisión del orden de las micras, su rango es del orden de uno o unos cuantos milímetros.

La desventaja de este tipo de sensores es que requieren de espacio libre para realizar la triangulación, a pesar de que la medición es puntual. Si existiera una pared alrededor del punto a medir puede esta desviar el rayo reflejado alterando o imposibilitando la medición.

### **b) Por Tiempo de Vuelo**

Este tipo de sensores es muy similar al sensor ultrasónico explicado anteriormente, con la diferencia que en vez de emplear pulsos ultrasónicos emplean pulsos de luz láser.

El gran problema de estos sensores es poder cuantificar de forma precisa el tiempo que tarda el rayo de luz en ir hacia la superficie y regresar ya que la luz viaja a una gran velocidad.

Estos sensores son por lo tanto poco precisos y solo son útiles para medir distancias grandes sin mucha precisión.

### **c) Por Corrimiento de Fase**

## **VII. Sensor**

---

Los sensores que utilizan el método de corrimiento de fase requieren de un haz de luz láser continuo y modulado.

La base de su funcionamiento es el de determinar el desfase entre el haz saliente y el haz entrante después de ser reflejado por la superficie.

Este desfase es linealmente proporcional a la distancia existente entre el sensor y la superficie reflejante. El desfase dependerá del tiempo que tarda el rayo en ir y venir desde el sensor a la superficie y de regreso, por lo que este tipo de sensor también es llamado “sensor por tiempo de vuelo”, pero utilizando un método totalmente diferente al descrito anteriormente.

Este tipo de sensores tienen una mejor precisión que la de los sensores ultrasónicos, pero no llega a igualar a la proporcionada por los sensores que utilizan el método de triangulación.

## **4. Conclusiones**

Debido a la amplia gama de aplicaciones donde se utilizan los sensores ultrasónicos y láseres, existe gran variedad de modelos en el mercado. Dependiendo del rango a medir, de la precisión requerida, del costo, etc.

Habiendo también sensores con diferentes tipos de interfaces para comunicarse con la computadora o microcontrolador encargado de gobernarlo. Entre estas interfaces están la RS-232, RS-422, RS-485, I<sup>2</sup>C, analógica.

Haciendo un análisis específico sobre cada uno de los sensores anteriormente descritos se tiene:

- Los sensores estereoscópicos no son prácticos debido a la gran complejidad (si es que se pudiera lograr) del procesamiento de las imágenes.
- Para nuestro diseño los sensores ultrasónicos no son útiles debido a su imposibilidad de realizar mediciones puntuales.
- Los sensores láser por tiempo de vuelo no cumplen con las necesidades de precisión establecidas.
- Los sensores láser por triangulación cumplen con las especificaciones solicitadas, pero es necesario tener en cuenta que para lograr la precisión adecuada se requieren sensores con

## VII. Sensor

---

un rango menor al necesario teniendo entonces que desplazar verticalmente al sensor para que la medición caiga dentro de ese rango. Otro punto importante es la necesidad del espacio libre para no interferir con la medición.

- El sensor láser por corrimiento de fase pudiera ser el más indicado para ser utilizado en el proyecto, pues brinda una medición puntual y permite contar con un mayor rango de medición, brindando la precisión necesaria y evitando el tener que desplazar en sentido vertical al sensor.

### 5. Sensor Propuesto

Para la realización de este prototipo se propone el modelo DT 200 de la marca Sick.

Ya que este sensor tiene un rango de 2m y una resolución de 1 mm, lo cual cumple con los requerimientos del prototipo.

La tabla de sus especificaciones se anexa en el capítulo referente a “Datos Técnicos”

## VIII. DISEÑO ELÉCTRICO-ELECTRÓNICO

---

En este capítulo se detallará el diseño Eléctrico-Electrónico de todas las partes involucradas en el sistema.

### 1. Fuentes de Alimentación

Para la correcta operación del sistema se precisa contar con tres niveles de voltaje diferentes: 5V, 24V y 30V.

La fuente de 5V se empleará para alimentar toda la parte digital del sistema, como son el microcontrolador, el generador de las secuencias para el giro de los motores y el convertidor de nivel para establecer el enlace entre la PC, el microcontrolador y los sensores de inicio y fin de carrera.

La fuente de 24V se utilizará para alimentar el sensor láser.

## VIII. Diseño Eléctrico-Electrónico

La fuente de 5 y la de 24V pueden estar construidas a partir del mismo transformador, aunque puede resultar más práctico construirlas por separado, ya que en caso contrario se necesitaría generar una caída de tensión de alrededor de 20V (para obtener los 5V a partir de los 24V) generando gran cantidad de calor por disipación de la energía y por lo tanto será necesario incluir radiadores y componentes con mayor capacidad de disipación de energía.

La fuente de 30V se utilizará para alimentar a los motores de pasos, debido a las características eléctricas de los motores, esta fuente requiere estar construida de forma independiente de las dos anteriores para evitar generar ruido eléctrico en las mismas.

Tanto La fuente de 5V como la de 24V requieren ser fuentes reguladas y de poco amperaje, no más de 1A.

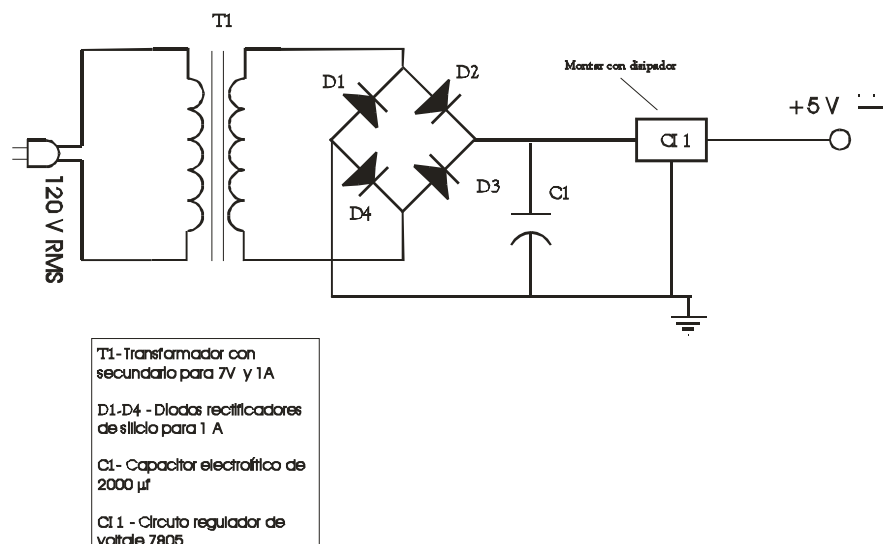


Fig. 11 .- Esquema de la fuente de 5 V



## VIII. Diseño Eléctrico-Electrónico

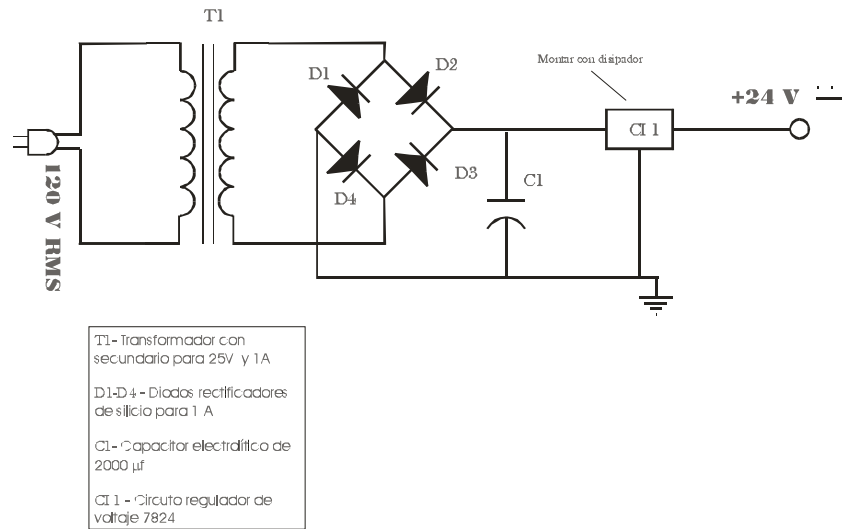


Fig. 12 .- Esquema de la fuente de 24 V

La fuente de 30V por el contrario no requiere ser una fuente regulada, pero si de mayor amperaje, de 4A.

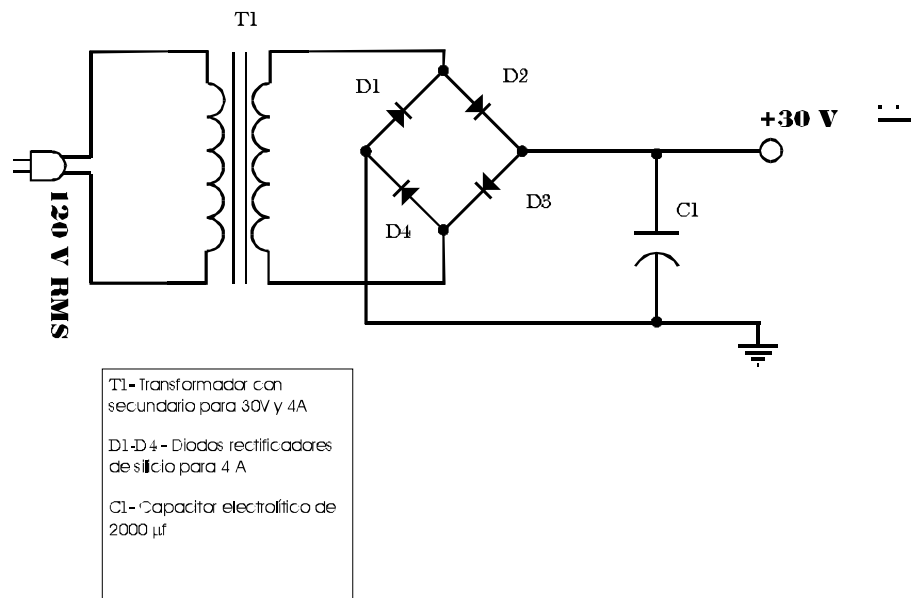


Fig. 13 .- Esquema de la fuente de 30 V

## VIII. Diseño Eléctrico-Electrónico

---

### 2. Sensores de Inicio y Fin de Carrera

Para poder determinar cuando el carro se encuentra en el inicio de su trayectoria (origen) o en el fin de la misma se requiere que en cada eje de desplazamiento se localicen sensores de inicio y fin de carrera.

Los sensores propuestos para este fin son interruptores magnéticos NA, los cuales al estar próximos al carro se activarán cerrando sus contactos y permitiendo que la señal de 5V (1 lógico) llegue a un puerto del microcontrolador.

El diagrama de conexión para cada uno de estos sensores es el siguiente:

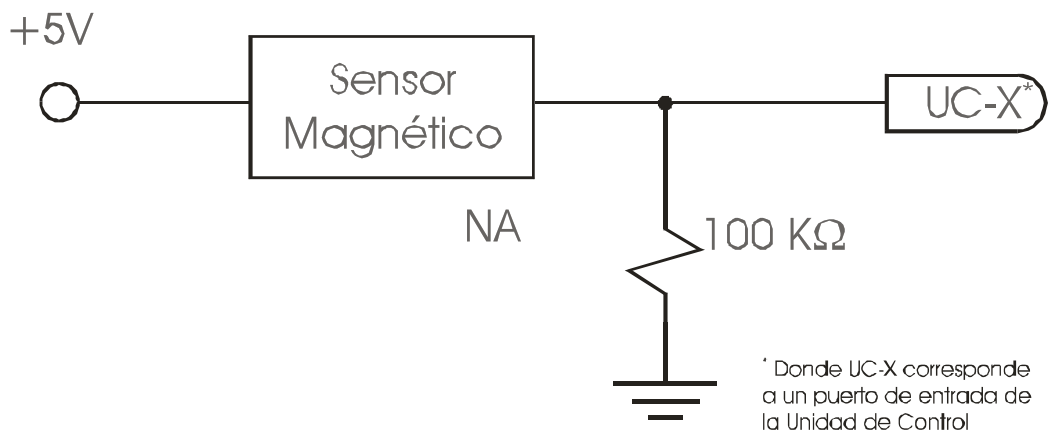


Fig. 14 .- Esquema eléctrico de los sensores de inicio y fin de carrera

Adicional a estos sensores se colocarán sensores mecánicos NC en serie con la fuente de 30V, los cuales serán activados por los ejes al intentar desplazarse más allá de los límites permitidos. Esto con la finalidad de

impedir que un mal funcionamiento o “congelamiento” del sistema de control tenga como consecuencia el forzar y dañar al sistema mecánico.

### 3. Enlace PC $\leftarrow \rightarrow$ Unidad de Control

Para lograr establecer el enlace entre la PC y la unidad de control se requiere de un convertidor de nivel encargado de convertir las señales provenientes del microcontrolador que conforma la Unidad de Control que son de 0 y 5V a señales compatibles con el estándar RS-232 que son señales de -10 y 10V.

Para lograr esto se utilizará el circuito integrado MAX232, que está diseñado para este fin.

Su función es a partir de lo 5V de la fuente duplicar dicho voltaje e invertirlo obteniendo los -10 y 10 V.

Este circuito integrado cuenta principalmente con 2 canales con 2 entradas y 2 salidas cada uno.

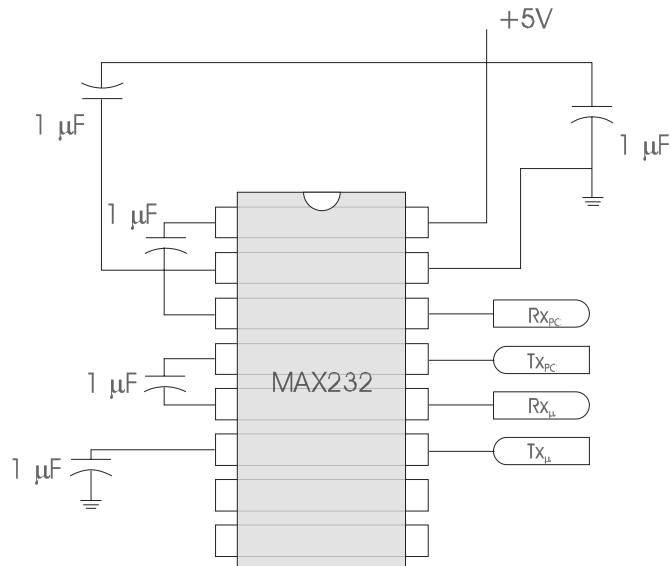
Por la entrada conectada del lado de la PC entrarán señales de  $\pm 10V$  y por la salida conectada del lado del microcontrolador saldrán señales de 0 o 5V de acuerdo al caso.

De manera recíproca sucederá con la otra entrada y salida.

## VIII. Diseño Eléctrico-Electrónico

---

El diagrama eléctrico es el siguiente.



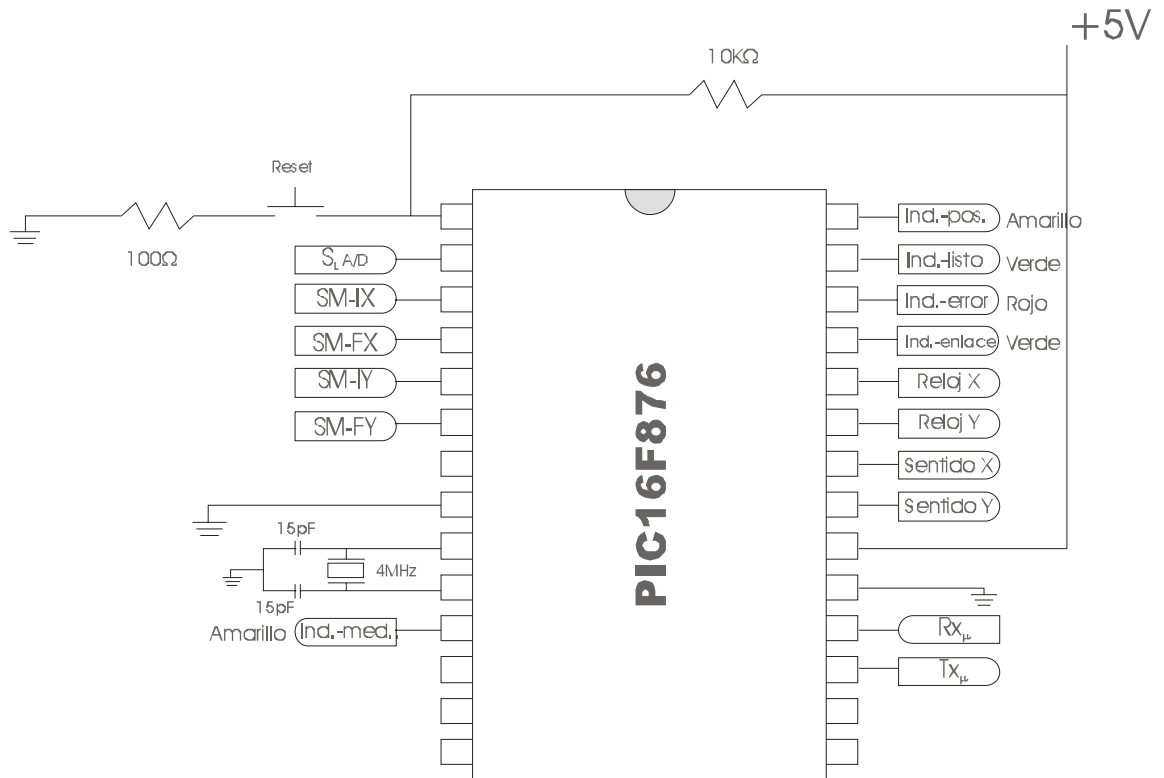
### 4. Unidad de Control

La unidad de Control estará constituida principalmente por un PIC 16F876.

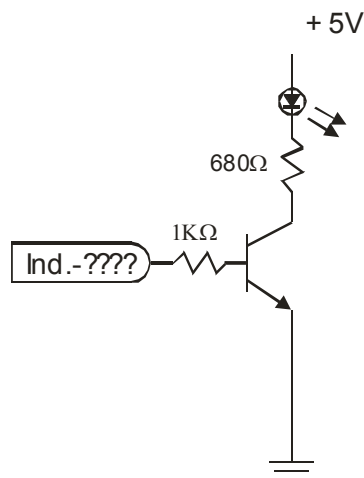
Para el funcionamiento de este microcontrolador se requiere agregar los componentes externos encargados de generar la señal de reloj apropiada, que en nuestro caso será de 4 MHz y el sistema de inicialización.

## VIII. Diseño Eléctrico-Electrónico

El diagrama eléctrico de dicha Unidad de Control será el siguiente:



En donde cada indicador (Ind-???) estará constituido de la forma siguiente:



### **5. Controladores para los Motores de Pasos**

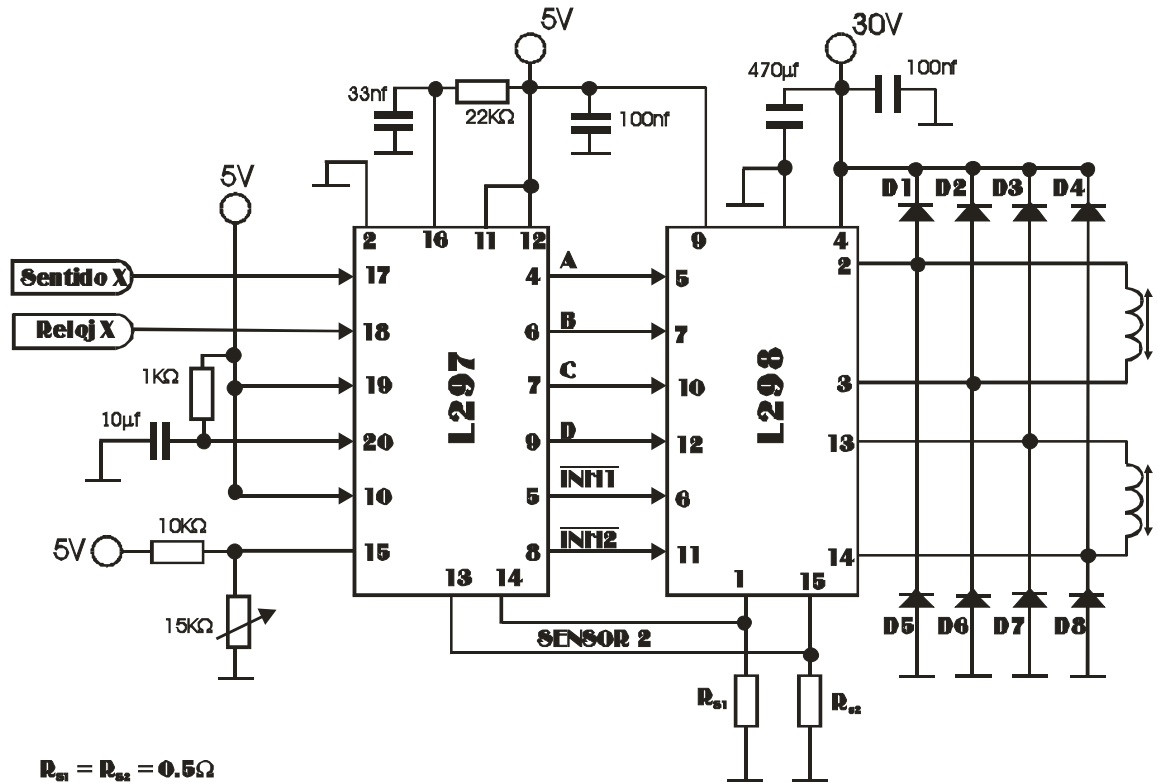
Para lograr el control de los motores de pasos se utilizarán dos circuitos integrados por motor, el LS297 y el LS298.

El LS297 es el encargado de generar las secuencias necesarias para hacer girar el motor en el sentido deseado y a la velocidad deseada. Para ello requiere que mediante un estado lógico (0 o 5V) le indiquemos el sentido del giro, y le proporcionemos una señal de reloj, cada flanco de bajada generará la señal necesaria para desplazar un paso al motor.

El LS298 será el encargado de proporcionar la potencia necesaria para que a partir de las señales generadas por el LS297 sea posible excitar las bobinas de los motores.

## VIII. Diseño Eléctrico-Electrónico

El diagrama de conexión es el siguiente:



$$R_{s1} = R_{s2} = 0.5\Omega$$

**D1 a D8 = 2 A Diodos Rápidos**  $\left\{ \begin{array}{l} V_F \leq 1.2V @ I = 2A \\ T_{rr} \leq 200 ns \end{array} \right.$





## IX. MEJORAS OPCIONALES

---

En esta sección se explicarán de forma breve algunas sugerencias sobre mejoras al equipo. Mejoras que dados los requerimientos de diseño no son esenciales, pero que si se implementaran llegarían a mejorar la funcionalidad del sistema.

### **1. Visión del modelo desde la perspectiva del sensor**

Esta mejora se refiere en montar en la cabeza sensora una mini cámara de video, que transmita la imagen captada hacia la PC, donde el usuario pueda visualizarla dentro de una ventana en el programa.

La ventaja de contar con esta mejora es que el usuario podría posicionar de forma visual el carro para realizar la medición sobre algún punto en

## **IX. Mejoras Opcionales**

---

particular. O tomar dicha decisión al ver cierta área de interés durante la realización de otro proceso de medición.

Dadas las características de construcción del modelo y de sus dimensiones es poco práctico e incluso imposible situar la Interfaz con el Usuario de manera que este pueda observar al modelo en su totalidad. Con esta mejora dicho problema no queda solucionado en su totalidad, pero no hay duda de que se reduciría en gran medida.

### **2. Panel manual autónomo**

El panel manual autónomo consistiría en un panel donde el usuario podría leer la posición del carro y el valor de la medición realizada mediante el uso de displays, y posicionar a voluntad el carro para realizar la medición en el punto que considere apropiado.

Este panel sería autónomo ya que no requeriría el uso de la computadora, pero con el inconveniente de tener que realizar medición por medición.

Esta modificación requeriría de grandes ajustes en todo el sistema y su funcionalidad es escasa. Una de las ventajas de este módulo radicaría en poder operar el sistema sin una PC, en el caso de que no se cuente con una, problema que en la actualidad es relativamente fácil de solucionar.

La otra ventaja sería que este módulo fuera móvil, permitiendo así que el operador se desplazara para tener un mejor punto de apreciación del modelo.

La accesibilidad al modelo es escasa debido a su construcción, y en realidad este panel no sería muy eficiente, por ello es que no se incluyó en el diseño del prototipo, pero se deja abierta la posibilidad de su implementación.

### **3. Modo de operación “Experto”**

En este modo de operación el programa de la PC tendría que decidir en función de parámetros introducidos por el usuario si se requiere que en alguna región se realicen más mediciones que las solicitadas inicialmente por el usuario (modo automático) dependiendo de los valores de las lecturas obtenidas.

Un criterio para tomar estas decisiones podría ser el calcular la pendiente de la recta que una dos puntos contiguos, y si el valor absoluto de esta es mayor que cierto parámetro facilitado por el usuario entonces se tomaría la lectura del punto equidistante a ambos puntos.

## **IX. Mejoras Opcionales**

---

Con este criterio se podría identificar cuando entre dos puntos medidos exista un cambio brusco de altura, tomando una lectura entre ambos para tratar de saber la causa de dicha diferencia.

### **4. Representación tridimensional del modelo**

El programa propuesto para este sistema ofrece la posibilidad de representar en forma tridimensional los planos de medición, uniendo los puntos contiguos con líneas rectas. Lo cual es poco probable que represente la realidad. Además esta representación gráfica se ve segmentada al realizar la gráfica por planos, uniendo los puntos solo en el sentido de un eje.

Dadas las condiciones actuales del diseño sería muy irresponsable escoger algún método de interpolación para generar la curva que una a dos puntos; pues es difícil predecir el comportamiento del fondo móvil del modelo, pudiendo tener cambios bruscos o suaves.

Sin embargo, haciendo uso del “modo experto” y escogiendo cuidadosamente los parámetros para realizar la medición es posible que se pueda tener un modelo tridimensional lo suficientemente representativo del modelo real.

Para poder tomar una buena decisión sobre esta parte del diseño habría que analizar el comportamiento del modelo y la funcionalidad de contar con una representación gráfica tridimensional del mismo.

Es por ello que solo se plantea la posibilidad de generar dicha representación y no se incluye como parte del diseño.

### **5. Graficación de “Curvas de Nivel”**

Esta mejora estaría muy ligada a la representación tridimensional del modelo y por ende al uso del “modo experto”. Es verdad que no se necesita la representación tridimensional para generar las curvas de nivel, pero si es necesario realizar la interpolación tridimensional entre los puntos analizados con la finalidad de poder establecer los niveles.



## X. ANEXO

---

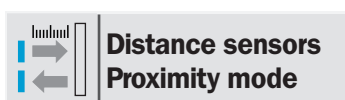
# DT 200: Laser analog sensor that measures distances up to 2000 mm



The sturdy metal housing with M 12 connector makes it possible to use the equipment even in difficult environmental conditions.

Typical applications are:

- sag (loop) control,
- contour detection,
- diameter detection of coils, for example,
- difference measurements for determining the thickness of parts.



**Distance sensors**  
**Proximity mode**

**D** DT 200 – Distance sensor for measurement distances up to 2 m.


The DT 200, which operates according to the principle of light run-time measurement, provides a resolution of 1 mm via a precise, analog power output.

The red measurement laser enables precise alignment to an object.

Small parts are detected precisely even at a distance of 2 m thanks to the small light spot, which is similar to that of a laser pointer.

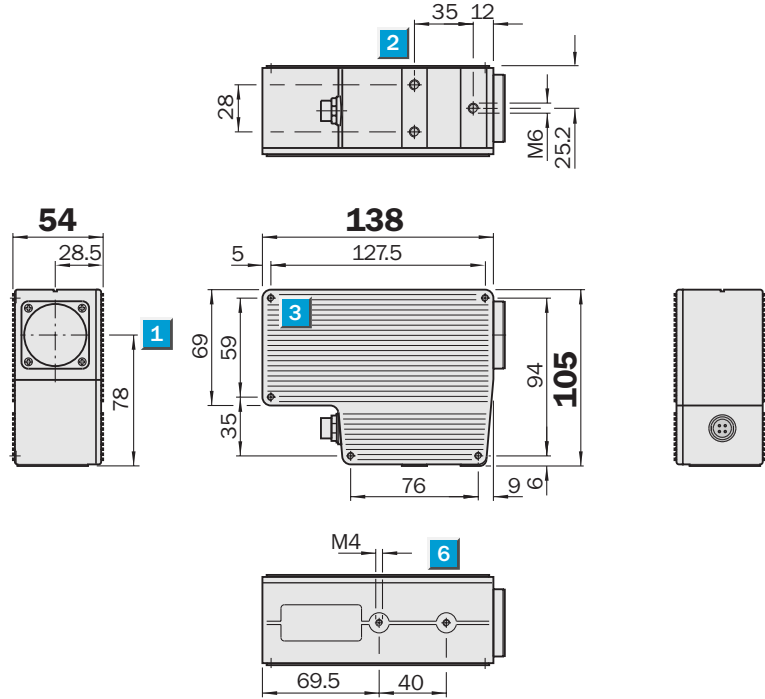
# SICK




**Measurement range**  
**100...2000 mm**  
**Distance Sensor**

- Good measurement accuracy thanks to of-flight measurement
- Analog output
- Easy adjustment using red laserlight
- Eye-save, laser class 2

**Dimensional drawing**

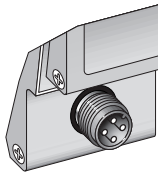


- 1 Centre of optical axis
- 2 M 6 threaded mounting hole, 10 mm deep
- 3 M 4 threaded mounting hole, 14 mm deep (this side only)
- 4 M 4 threaded mounting hole, 6 mm deep

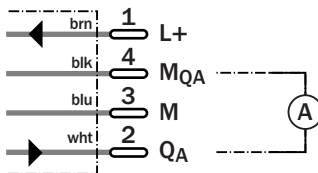


**Connection type**

DT 200-111



4-pin, M 12



Laser class 2

Technical Data		DT 200-	111	111A01							
<b>Measurement range<sup>1)</sup></b>	100 ... 2000 mm										
<b>Accuracy</b>											
Object with 90 % remission	± 5 mm										
Object with 18 % remission	± 11 mm										
Object with 6 % remission	± 65 mm										
<b>Reproducibility</b>											
Object with 90 % remission	1 mm										
Object with 18 % remission	3 mm										
Object with 6 % remission	25 mm										
<b>Light source<sup>2)</sup>, light type</b>	Laser diode, red light										
Laser category	2 (IEC 825-1/EN 60825-1)										
	CDRH										
<b>Supply voltage V<sub>s</sub></b>	18 ... 30 V DC <sup>3)</sup>										
Ripple <sup>4)</sup>	< 5 V <sub>pp</sub>										
Power consumption <sup>5)</sup>	≤ 6 W										
<b>Analog output</b>	0 ... 20 mA (0.01 mA/mm)										
Response time	< 30 ms										
Temperature drift	0.4 mm/K										
<b>Connection type</b>	M 12 plug, 4-pin										
<b>VDE protection class<sup>6)</sup></b>	<input type="checkbox"/>										
<b>Enclosure rating</b>	IP 67										
<b>Ambient temperature T<sub>A</sub></b>	Operation – 10 °C ... + 45 °C										
	Storage – 25 °C ... + 75 °C										
<b>Weight</b>	Approx. 980 g										

<sup>1)</sup> Falling below of the measurement range results in values < 0.1 mA  
Exceeding the measurement range results in ambiguous values  
(Example: Scanning distance of 3 m  $\hat{=}$  10 mA with sufficient reflectance)

<sup>2)</sup> Average service life 50,000 h at T<sub>A</sub> = + 25 °C  
<sup>3)</sup> Limit values, reverse-polarity protected

<sup>4)</sup> May not exceed or fall short of V<sub>s</sub> tolerances  
<sup>5)</sup> Without load  
<sup>6)</sup> Reference voltage DC 50 V

Order Information	
Type	Part no.
DT 200-111	1 022 938
DT 200-111A01	1 022 946

## 28/40-Pin 8-Bit CMOS FLASH Microcontrollers

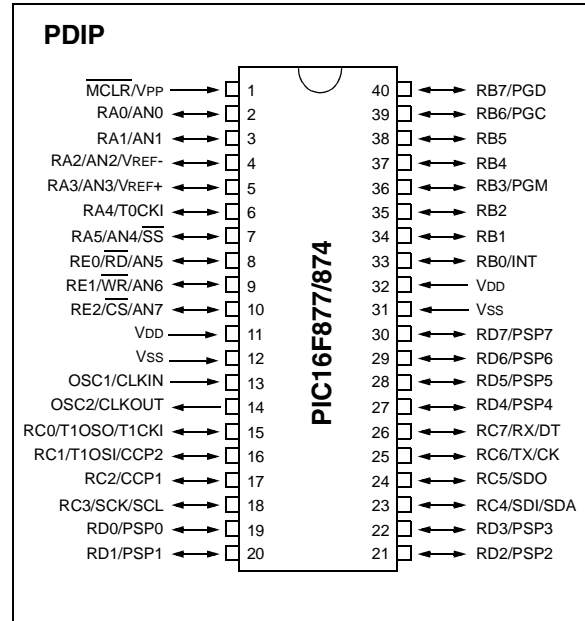
### Devices Included in this Data Sheet:

- PIC16F873
- PIC16F876
- PIC16F874
- PIC16F877

### Microcontroller Core Features:

- High performance RISC CPU
- Only 35 single word instructions to learn
- All single cycle instructions except for program branches which are two cycle
- Operating speed: DC - 20 MHz clock input  
DC - 200 ns instruction cycle
- Up to 8K x 14 words of FLASH Program Memory,  
Up to 368 x 8 bytes of Data Memory (RAM)  
Up to 256 x 8 bytes of EEPROM Data Memory
- Pinout compatible to the PIC16C73B/74B/76/77
- Interrupt capability (up to 14 sources)
- Eight level deep hardware stack
- Direct, indirect and relative addressing modes
- Power-on Reset (POR)
- Power-up Timer (PWRT) and  
Oscillator Start-up Timer (OST)
- Watchdog Timer (WDT) with its own on-chip RC  
oscillator for reliable operation
- Programmable code protection
- Power saving SLEEP mode
- Selectable oscillator options
- Low power, high speed CMOS FLASH/EEPROM  
technology
- Fully static design
- In-Circuit Serial Programming™ (ICSP) via two  
pins
- Single 5V In-Circuit Serial Programming capability
- In-Circuit Debugging via two pins
- Processor read/write access to program memory
- Wide operating voltage range: 2.0V to 5.5V
- High Sink/Source Current: 25 mA
- Commercial, Industrial and Extended temperature  
ranges
- Low-power consumption:
  - < 0.6 mA typical @ 3V, 4 MHz
  - 20 µA typical @ 3V, 32 kHz
  - < 1 µA typical standby current

### Pin Diagram



### Peripheral Features:

- Timer0: 8-bit timer/counter with 8-bit prescaler
- Timer1: 16-bit timer/counter with prescaler,  
can be incremented during SLEEP via external  
crystal/clock
- Timer2: 8-bit timer/counter with 8-bit period  
register, prescaler and postscaler
- Two Capture, Compare, PWM modules
  - Capture is 16-bit, max. resolution is 12.5 ns
  - Compare is 16-bit, max. resolution is 200 ns
  - PWM max. resolution is 10-bit
- 10-bit multi-channel Analog-to-Digital converter
- Synchronous Serial Port (SSP) with SPI™ (Master  
mode) and I<sup>2</sup>C™ (Master/Slave)
- Universal Synchronous Asynchronous Receiver  
Transmitter (USART/SCI) with 9-bit address  
detection
- Parallel Slave Port (PSP) 8-bits wide, with  
external  $\overline{RD}$ ,  $\overline{WR}$  and  $\overline{CS}$  controls (40/44-pin only)
- Brown-out detection circuitry for  
Brown-out Reset (BOR)

**FIGURE 2-3: PIC16F877/876 REGISTER FILE MAP**

File Address		File Address		File Address		File Address			
Indirect addr. <sup>(*)</sup>	00h	Indirect addr. <sup>(*)</sup>	80h	Indirect addr. <sup>(*)</sup>	100h	Indirect addr. <sup>(*)</sup>	180h		
TMR0	01h	OPTION_REG	81h	TMR0	101h	OPTION_REG	181h		
PCL	02h	PCL	82h	PCL	102h	PCL	182h		
STATUS	03h	STATUS	83h	STATUS	103h	STATUS	183h		
FSR	04h	FSR	84h	FSR	104h	FSR	184h		
PORTA	05h	TRISA	85h		105h		185h		
PORTB	06h	TRISB	86h	PORTB	106h	TRISB	186h		
PORTC	07h	TRISC	87h		107h		187h		
PORTD <sup>(1)</sup>	08h	TRISD <sup>(1)</sup>	88h		108h		188h		
PORTE <sup>(1)</sup>	09h	TRISE <sup>(1)</sup>	89h		109h		189h		
PCLATH	0Ah	PCLATH	8Ah	PCLATH	10Ah	PCLATH	18Ah		
INTCON	0Bh	INTCON	8Bh	INTCON	10Bh	INTCON	18Bh		
PIR1	0Ch	PIE1	8Ch	EEDATA	10Ch	EECON1	18Ch		
PIR2	0Dh	PIE2	8Dh	EEADR	10Dh	EECON2	18Dh		
TMR1L	0Eh	PCON	8Eh	EEDATH	10Eh	Reserved <sup>(2)</sup>	18Eh		
TMR1H	0Fh		8Fh	EEADRH	10Fh	Reserved <sup>(2)</sup>	18Fh		
T1CON	10h		90h	General Purpose Register 16 Bytes		General Purpose Register 16 Bytes		190h	
TMR2	11h	SSPCON2	91h				110h		191h
T2CON	12h	PR2	92h				111h		192h
SSPBUF	13h	SSPADD	93h				112h		193h
SSPCON	14h	SSPSTAT	94h				113h		194h
CCPR1L	15h		95h				114h		195h
CCPR1H	16h		96h				115h		196h
CCP1CON	17h		97h				116h		197h
RCSTA	18h	TXSTA	98h				117h		198h
TXREG	19h	SPBRG	99h				118h		199h
RCREG	1Ah		9Ah				119h		19Ah
CCPR2L	1Bh		9Bh				11Ah		19Bh
CCPR2H	1Ch		9Ch				11Bh		19Ch
CCP2CON	1Dh		9Dh				11Ch		19Dh
ADRESH	1Eh	ADRESL	9Eh				11Dh		19Dh
ADCON0	1Fh	ADCON1	9Fh				11Eh		19Eh
General Purpose Register 96 Bytes	20h		A0h		11Fh		19Fh		
	General Purpose Register 80 Bytes	General Purpose Register 80 Bytes				120h	General Purpose Register 80 Bytes		1A0h
		accesses 70h-7Fh	EFh F0h	accesses 70h-7Fh	16Fh 170h	accesses 70h - 7Fh	1EFh 1F0h		
	7Fh		FFh		17Fh		1FFh		
Bank 0		Bank 1		Bank 2		Bank 3			

Unimplemented data memory locations, read as '0'.  
 \* Not a physical register.

**Note 1:** These registers are not implemented on the PIC16F876.  
**Note 2:** These registers are reserved, maintain these registers clear.

**TABLE 1-1: PIC16F873 AND PIC16F876 PINOUT DESCRIPTION**

Pin Name	DIP Pin#	SOIC Pin#	I/O/P Type	Buffer Type	Description
OSC1/CLKIN	9	9	I	ST/CMOS <sup>(3)</sup>	Oscillator crystal input/external clock source input.
OSC2/CLKOUT	10	10	O	—	Oscillator crystal output. Connects to crystal or resonator in crystal oscillator mode. In RC mode, the OSC2 pin outputs CLKOUT which has 1/4 the frequency of OSC1, and denotes the instruction cycle rate.
MCLR/VPP	1	1	I/P	ST	Master Clear (Reset) input or programming voltage input. This pin is an active low RESET to the device.
RA0/AN0	2	2	I/O	TTL	<p>PORTA is a bi-directional I/O port.</p> <p>RA0 can also be analog input0.</p> <p>RA1 can also be analog input1.</p> <p>RA2 can also be analog input2 or negative analog reference voltage.</p> <p>RA3 can also be analog input3 or positive analog reference voltage.</p> <p>RA4 can also be the clock input to the Timer0 module. Output is open drain type.</p> <p>RA5 can also be analog input4 or the slave select for the synchronous serial port.</p>
RA1/AN1	3	3	I/O	TTL	
RA2/AN2/VREF-	4	4	I/O	TTL	
RA3/AN3/VREF+	5	5	I/O	TTL	
RA4/T0CKI	6	6	I/O	ST	
RA5/SS/AN4	7	7	I/O	TTL	
RB0/INT	21	21	I/O	TTL/ST <sup>(1)</sup>	<p>PORTB is a bi-directional I/O port. PORTB can be software programmed for internal weak pull-up on all inputs.</p> <p>RB0 can also be the external interrupt pin.</p> <p>RB3 can also be the low voltage programming input.</p> <p>Interrupt-on-change pin.</p> <p>Interrupt-on-change pin.</p> <p>Interrupt-on-change pin or In-Circuit Debugger pin. Serial programming clock.</p> <p>Interrupt-on-change pin or In-Circuit Debugger pin. Serial programming data.</p>
RB1	22	22	I/O	TTL	
RB2	23	23	I/O	TTL	
RB3/PGM	24	24	I/O	TTL	
RB4	25	25	I/O	TTL	
RB5	26	26	I/O	TTL	
RB6/PGC	27	27	I/O	TTL/ST <sup>(2)</sup>	
RB7/PGD	28	28	I/O	TTL/ST <sup>(2)</sup>	
RC0/T1OSO/T1CKI	11	11	I/O	ST	<p>PORTC is a bi-directional I/O port.</p> <p>RC0 can also be the Timer1 oscillator output or Timer1 clock input.</p> <p>RC1 can also be the Timer1 oscillator input or Capture2 input/Compare2 output/PWM2 output.</p> <p>RC2 can also be the Capture1 input/Compare1 output/PWM1 output.</p> <p>RC3 can also be the synchronous serial clock input/output for both SPI and I<sup>2</sup>C modes.</p> <p>RC4 can also be the SPI Data In (SPI mode) or data I/O (I<sup>2</sup>C mode).</p> <p>RC5 can also be the SPI Data Out (SPI mode).</p> <p>RC6 can also be the USART Asynchronous Transmit or Synchronous Clock.</p> <p>RC7 can also be the USART Asynchronous Receive or Synchronous Data.</p>
RC1/T1OSI/CCP2	12	12	I/O	ST	
RC2/CCP1	13	13	I/O	ST	
RC3/SCK/SCL	14	14	I/O	ST	
RC4/SDI/SDA	15	15	I/O	ST	
RC5/SDO	16	16	I/O	ST	
RC6/TX/CK	17	17	I/O	ST	
RC7/RX/DT	18	18	I/O	ST	
VSS	8, 19	8, 19	P	—	Ground reference for logic and I/O pins.
VDD	20	20	P	—	Positive supply for logic and I/O pins.

Legend: I = input    O = output    I/O = input/output    P = power  
 — = Not used    TTL = TTL input    ST = Schmitt Trigger input

- Note 1:** This buffer is a Schmitt Trigger input when configured as the external interrupt.  
**Note 2:** This buffer is a Schmitt Trigger input when used in Serial Programming mode.  
**Note 3:** This buffer is a Schmitt Trigger input when configured in RC oscillator mode and a CMOS input otherwise.

<b>Key Features PICmicro™ Mid-Range Reference Manual (DS33023)</b>	<b>PIC16F873</b>	<b>PIC16F874</b>	<b>PIC16F876</b>	<b>PIC16F877</b>
Operating Frequency	DC - 20 MHz	DC - 20 MHz	DC - 20 MHz	DC - 20 MHz
RESETS (and Delays)	POR, BOR (PWRT, OST)	POR, BOR (PWRT, OST)	POR, BOR (PWRT, OST)	POR, BOR (PWRT, OST)
FLASH Program Memory (14-bit words)	4K	4K	8K	8K
Data Memory (bytes)	192	192	368	368
EEPROM Data Memory	128	128	256	256
Interrupts	13	14	13	14
I/O Ports	Ports A,B,C	Ports A,B,C,D,E	Ports A,B,C	Ports A,B,C,D,E
Timers	3	3	3	3
Capture/Compare/PWM Modules	2	2	2	2
Serial Communications	MSSP, USART	MSSP, USART	MSSP, USART	MSSP, USART
Parallel Communications	—	PSP	—	PSP
10-bit Analog-to-Digital Module	5 input channels	8 input channels	5 input channels	8 input channels
Instruction Set	35 instructions	35 instructions	35 instructions	35 instructions

# PIC16F87X

TABLE 13-2: PIC16F87X INSTRUCTION SET

Mnemonic, Operands	Description	Cycles	14-Bit Opcode				Status Affected	Notes	
			MSb		LSb				
<b>BYTE-ORIENTED FILE REGISTER OPERATIONS</b>									
ADDWF	f, d	Add W and f	1	00	0111	dfff	ffff	C,DC,Z	1,2
ANDWF	f, d	AND W with f	1	00	0101	dfff	ffff	Z	1,2
CLRF	f	Clear f	1	00	0001	1fff	ffff	Z	2
CLRWF	-	Clear W	1	00	0001	0xxx	xxxx	Z	
COMF	f, d	Complement f	1	00	1001	dfff	ffff	Z	1,2
DECf	f, d	Decrement f	1	00	0011	dfff	ffff	Z	1,2
DECFSZ	f, d	Decrement f, Skip if 0	1(2)	00	1011	dfff	ffff		1,2,3
INCF	f, d	Increment f	1	00	1010	dfff	ffff	Z	1,2
INCFSSZ	f, d	Increment f, Skip if 0	1(2)	00	1111	dfff	ffff		1,2,3
IORWF	f, d	Inclusive OR W with f	1	00	0100	dfff	ffff	Z	1,2
MOVF	f, d	Move f	1	00	1000	dfff	ffff	Z	1,2
MOVWF	f	Move W to f	1	00	0000	1fff	ffff		
NOP	-	No Operation	1	00	0000	0xx0	0000		
RLF	f, d	Rotate Left f through Carry	1	00	1101	dfff	ffff	C	1,2
RRF	f, d	Rotate Right f through Carry	1	00	1100	dfff	ffff	C	1,2
SUBWF	f, d	Subtract W from f	1	00	0010	dfff	ffff	C,DC,Z	1,2
SWAPF	f, d	Swap nibbles in f	1	00	1110	dfff	ffff		1,2
XORWF	f, d	Exclusive OR W with f	1	00	0110	dfff	ffff	Z	1,2
<b>BIT-ORIENTED FILE REGISTER OPERATIONS</b>									
BCF	f, b	Bit Clear f	1	01	00bb	bfff	ffff		1,2
BSF	f, b	Bit Set f	1	01	01bb	bfff	ffff		1,2
BTFSC	f, b	Bit Test f, Skip if Clear	1 (2)	01	10bb	bfff	ffff		3
BTFSS	f, b	Bit Test f, Skip if Set	1 (2)	01	11bb	bfff	ffff		3
<b>LITERAL AND CONTROL OPERATIONS</b>									
ADDLW	k	Add literal and W	1	11	111x	kkkk	kkkk	C,DC,Z	
ANDLW	k	AND literal with W	1	11	1001	kkkk	kkkk	Z	
CALL	k	Call subroutine	2	10	0kkk	kkkk	kkkk		
CLRWDTP	-	Clear Watchdog Timer	1	00	0000	0110	0100	$\overline{TO,PD}$	
GOTO	k	Go to address	2	10	1kkk	kkkk	kkkk		
IORLW	k	Inclusive OR literal with W	1	11	1000	kkkk	kkkk	Z	
MOVLW	k	Move literal to W	1	11	00xx	kkkk	kkkk		
RETFIE	-	Return from interrupt	2	00	0000	0000	1001		
RETLW	k	Return with literal in W	2	11	01xx	kkkk	kkkk		
RETURN	-	Return from Subroutine	2	00	0000	0000	1000		
SLEEP	-	Go into standby mode	1	00	0000	0110	0011	$\overline{TO,PD}$	
SUBLW	k	Subtract W from literal	1	11	110x	kkkk	kkkk	C,DC,Z	
XORLW	k	Exclusive OR literal with W	1	11	1010	kkkk	kkkk	Z	

- Note 1:** When an I/O register is modified as a function of itself ( e.g., MOVF PORTB, 1 ), the value used will be that value present on the pins themselves. For example, if the data latch is '1' for a pin configured as input and is driven low by an external device, the data will be written back with a '0'.
- 2:** If this instruction is executed on the TMR0 register (and, where applicable, d = 1), the prescaler will be cleared if assigned to the Timer0 module.
- 3:** If Program Counter (PC) is modified, or a conditional test is true, the instruction requires two cycles. The second cycle is executed as a NOP.

**Note:** Additional information on the mid-range instruction set is available in the PICmicro™ Mid-Range MCU Family Reference Manual (DS33023).



# +5V-Powered, Multichannel RS-232 Drivers/Receivers

## General Description

The MAX220–MAX249 family of line drivers/receivers is intended for all EIA/TIA-232E and V.28/V.24 communications interfaces, particularly applications where  $\pm 12V$  is not available.

These parts are especially useful in battery-powered systems, since their low-power shutdown mode reduces power dissipation to less than  $5\mu W$ . The MAX225, MAX233, MAX235, and MAX245/MAX246/MAX247 use no external components and are recommended for applications where printed circuit board space is critical.

## Applications

Portable Computers  
Low-Power Modems  
Interface Translation  
Battery-Powered RS-232 Systems  
Multidrop RS-232 Networks

## Next-Generation Device Features

- ◆ For Low-Voltage, Integrated ESD Applications: MAX3222E/MAX3232E/MAX3237E/MAX3241E/MAX3246E: +3.0V to +5.5V, Low-Power, Up to 1Mbps, True RS-232 Transceivers Using Four 0.1 $\mu F$  External Capacitors (MAX3246E Available in a UCSP™ Package)
- ◆ For Low-Cost Applications: MAX221E:  $\pm 15kV$  ESD-Protected, +5V, 1 $\mu A$ , Single RS-232 Transceiver with AutoShutdown™

## Ordering Information

PART	TEMP RANGE	PIN-PACKAGE
MAX220CPE	0°C to +70°C	16 Plastic DIP
MAX220CSE	0°C to +70°C	16 Narrow SO
MAX220CWE	0°C to +70°C	16 Wide SO
MAX220C/D	0°C to +70°C	Dice*
MAX220EPE	-40°C to +85°C	16 Plastic DIP
MAX220ESE	-40°C to +85°C	16 Narrow SO
MAX220EWE	-40°C to +85°C	16 Wide SO
MAX220EJE	-40°C to +85°C	16 CERDIP
MAX220MJE	-55°C to +125°C	16 CERDIP

Ordering Information continued at end of data sheet.

\*Contact factory for dice specifications.

## Selection Table

Part Number	Power Supply (V)	No. of RS-232 Drivers/Rx	No. of Ext. Caps	Nominal Cap. Value ( $\mu F$ )	SHDN & Three-State	Rx Active in SHDN	Data Rate (kbps)	Features
MAX220	+5	2/2	4	0.1	No	—	120	Ultra-low-power, industry-standard pinout
MAX222	+5	2/2	4	0.1	Yes	—	200	Low-power shutdown
MAX223 (MAX213)	+5	4/5	4	1.0 (0.1)	Yes	✓	120	MAX241 and receivers active in shutdown
MAX225	+5	5/5	0	—	Yes	✓	120	Available in SO
MAX230 (MAX200)	+5	5/0	4	1.0 (0.1)	Yes	—	120	5 drivers with shutdown
MAX231 (MAX201)	+5 and +7.5 to +13.2	2/2	2	1.0 (0.1)	No	—	120	Standard +5/+12V or battery supplies; same functions as MAX232
MAX232 (MAX202)	+5	2/2	4	1.0 (0.1)	No	—	120 (64)	Industry standard
MAX232A	+5	2/2	4	0.1	No	—	200	Higher slew rate, small caps
MAX233 (MAX203)	+5	2/2	0	—	No	—	120	No external caps
MAX233A	+5	2/2	0	—	No	—	200	No external caps, high slew rate
MAX234 (MAX204)	+5	4/0	4	1.0 (0.1)	No	—	120	Replaces 1488
MAX235 (MAX205)	+5	5/5	0	—	Yes	—	120	No external caps
MAX236 (MAX206)	+5	4/3	4	1.0 (0.1)	Yes	—	120	Shutdown, three state
MAX237 (MAX207)	+5	5/3	4	1.0 (0.1)	No	—	120	Complements IBM PC serial port
MAX238 (MAX208)	+5	4/4	4	1.0 (0.1)	No	—	120	Replaces 1488 and 1489
MAX239 (MAX209)	+5 and +7.5 to +13.2	3/5	2	1.0 (0.1)	No	—	120	Standard +5/+12V or battery supplies; single-package solution for IBM PC serial port
MAX240	+5	5/5	4	1.0	Yes	—	120	DIP or flatpack package
MAX241 (MAX211)	+5	4/5	4	1.0 (0.1)	Yes	—	120	Complete IBM PC serial port
MAX242	+5	2/2	4	0.1	Yes	✓	200	Separate shutdown and enable
MAX243	+5	2/2	4	0.1	No	—	200	Open-line detection simplifies cabling
MAX244	+5	8/10	4	1.0	No	—	120	High slew rate
MAX245	+5	8/10	0	—	Yes	✓	120	High slew rate, int. caps, two shutdown modes
MAX246	+5	8/10	0	—	Yes	✓	120	High slew rate, int. caps, three shutdown modes
MAX247	+5	8/9	0	—	Yes	✓	120	High slew rate, int. caps, nine operating modes
MAX248	+5	8/8	4	1.0	Yes	✓	120	High slew rate, selective half-chip enables
MAX249	+5	6/10	4	1.0	Yes	✓	120	Available in quad flatpack package





# +5V-Powered, Multichannel RS-232 Drivers/Receivers

**MAX220-MAX249**

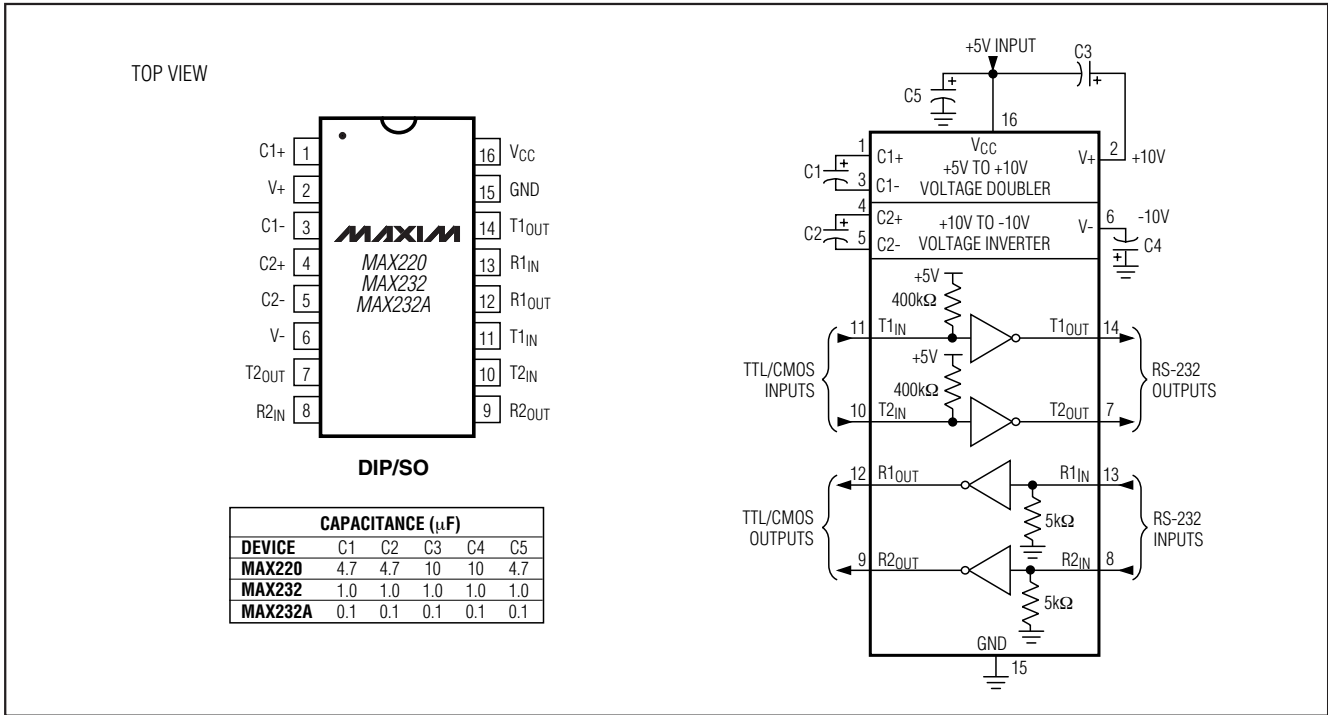
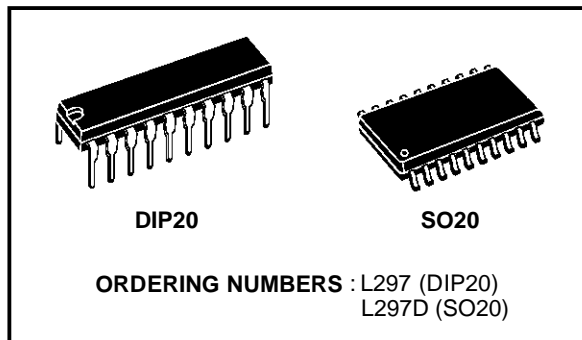


Figure 5. MAX220/MAX232/MAX232A Pin Configuration and Typical Operating Circuit

## STEPPER MOTOR CONTROLLERS

- NORMAL/WAVE DRIVE
- HALF/FULL STEP MODES
- CLOCKWISE/ANTICLOCKWISE DIRECTION
- SWITCHMODE LOAD CURRENT REGULATION
- PROGRAMMABLE LOAD CURRENT
- FEW EXTERNAL COMPONENTS
- RESET INPUT & HOME OUTPUT
- ENABLE INPUT



### DESCRIPTION

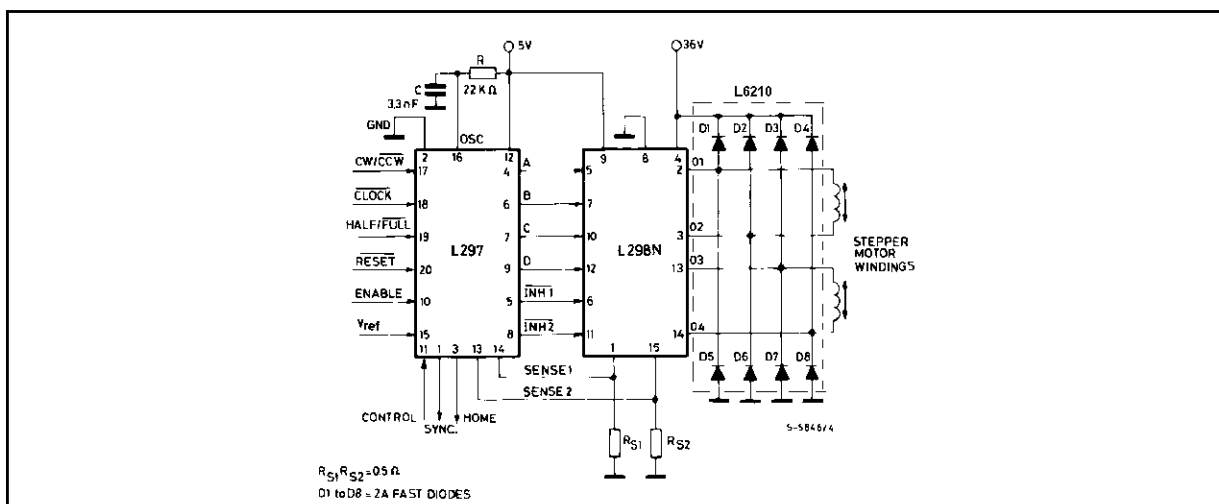
The L297/A/D Stepper Motor Controller IC generates four phase drive signals for two phase bipolar and four phase unipolar step motors in microcomputer-controlled applications. The motor can be driven in half step, normal and wave drive modes and on-chip PWM chopper circuits permit switchmode control of the current in the windings. A

feature of this device is that it requires only clock, direction and mode input signals. Since the phase are generated internally the burden on the microprocessor, and the programmer, is greatly reduced. Mounted in DIP20 and SO20 packages, the L297 can be used with monolithic bridge drives such as the L298N or L293E, or with discrete transistors and darlingtonts.

### ABSOLUTE MAXIMUM RATINGS

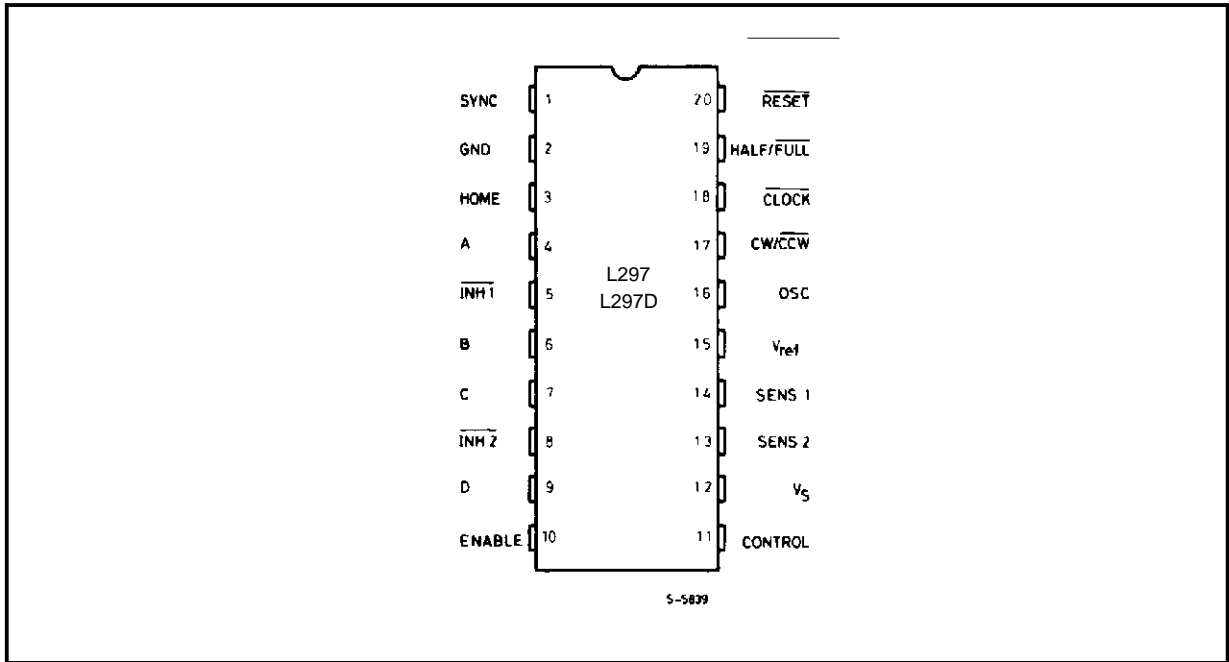
Symbol	Parameter	Value	Unit
$V_s$	Supply voltage	10	V
$V_i$	Input signals	7	V
$P_{tot}$	Total power dissipation ( $T_{amb} = 70^{\circ}C$ )	1	W
$T_{stg}, T_j$	Storage and junction temperature	-40 to + 150	$^{\circ}C$

### TWO PHASE BIPOLAR STEPPER MOTOR CONTROL CIRCUIT

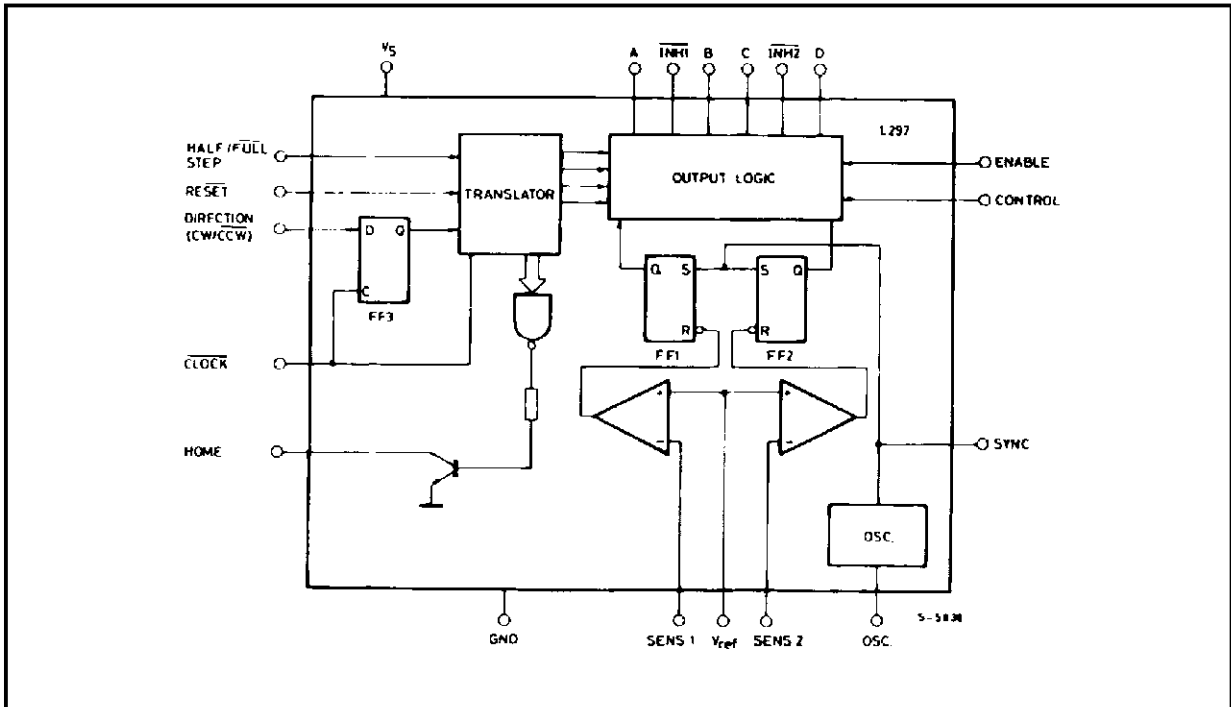


# L297-L297D

## PIN CONNECTION (Top view)



## BLOCK DIAGRAM (L297/L297D)

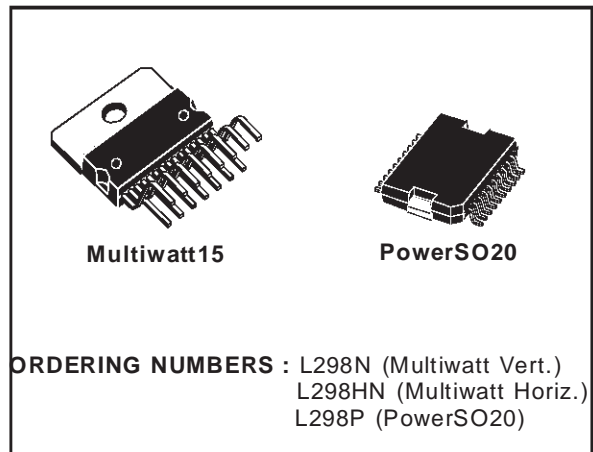


## DUAL FULL-BRIDGE DRIVER

- OPERATING SUPPLY VOLTAGE UP TO 46 V
- TOTAL DC CURRENT UP TO 4 A
- LOW SATURATION VOLTAGE
- OVERTEMPERATURE PROTECTION
- LOGICAL "0" INPUT VOLTAGE UP TO 1.5 V (HIGH NOISE IMMUNITY)

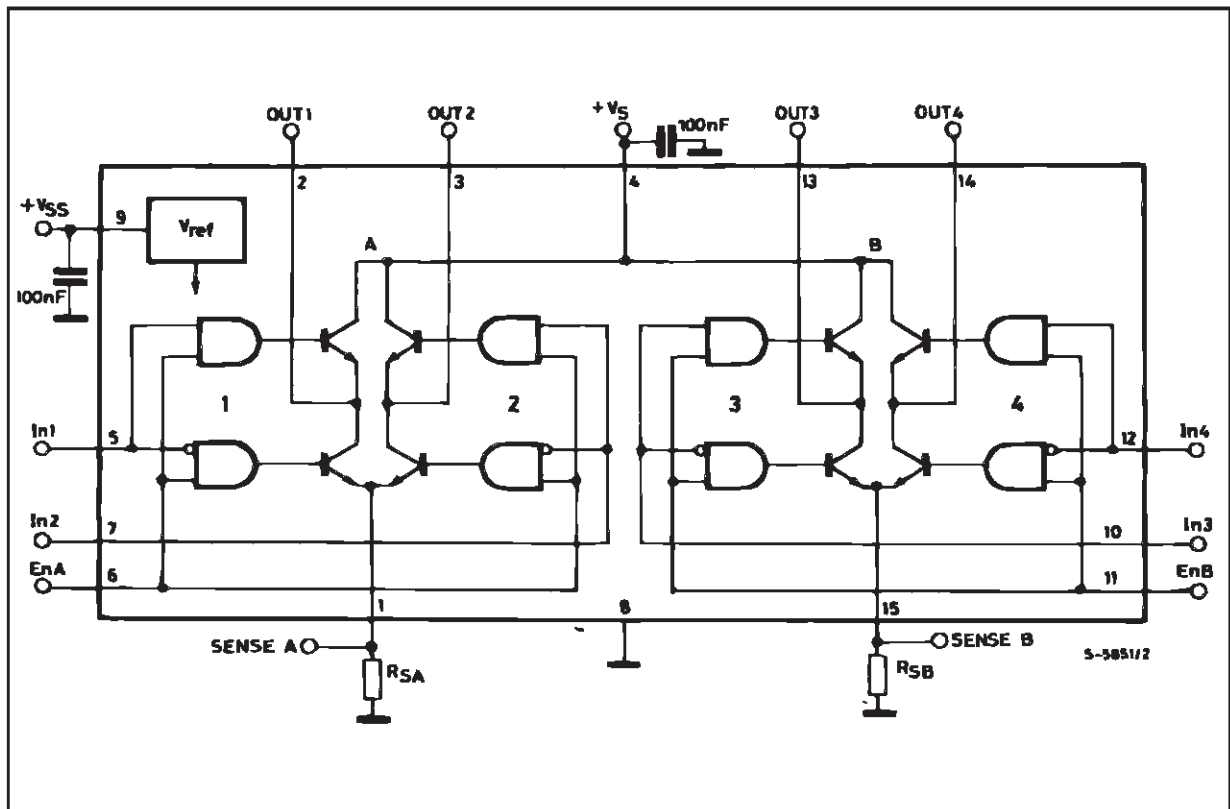
### DESCRIPTION

The L298 is an integrated monolithic circuit in a 15-lead Multiwatt and PowerSO20 packages. It is a high voltage, high current dual full-bridge driver designed to accept standard TTL logic levels and drive inductive loads such as relays, solenoids, DC and stepping motors. Two enable inputs are provided to enable or disable the device independently of the input signals. The emitters of the lower transistors of each bridge are connected together and the corresponding external terminal can be used for the con-



nection of an external sensing resistor. An additional supply input is provided so that the logic works at a lower voltage.

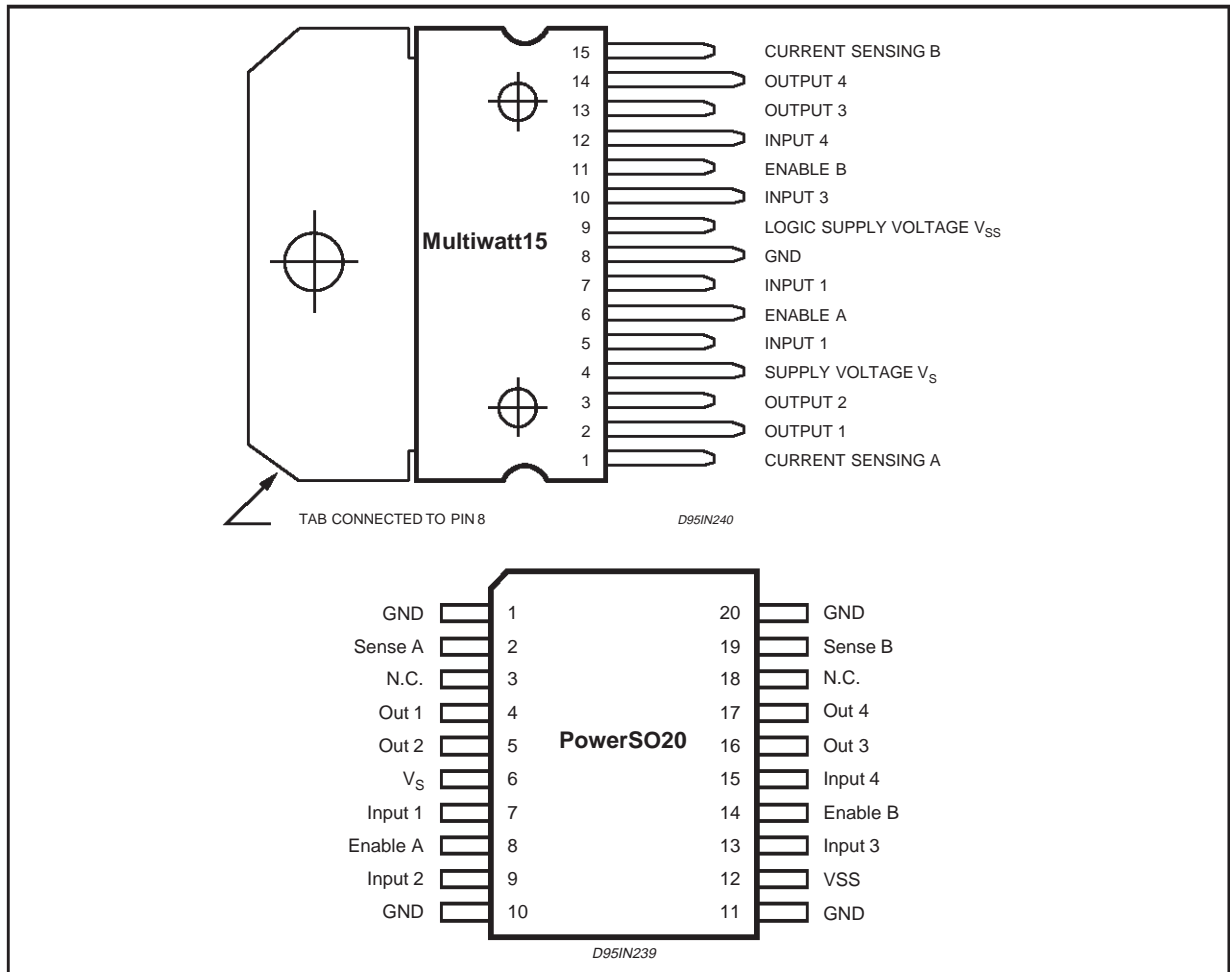
### BLOCK DIAGRAM



**ABSOLUTE MAXIMUM RATINGS**

Symbol	Parameter	Value	Unit
$V_S$	Power Supply	50	V
$V_{SS}$	Logic Supply Voltage	7	V
$V_I, V_{en}$	Input and Enable Voltage	-0.3 to 7	V
$I_O$	Peak Output Current (each Channel)		
	- Non Repetitive ( $t = 100\mu s$ )	3	A
	- Repetitive (80% on -20% off; $t_{on} = 10ms$ )	2.5	A
	-DC Operation	2	A
$V_{sens}$	Sensing Voltage	-1 to 2.3	V
$P_{tot}$	Total Power Dissipation ( $T_{case} = 75^\circ C$ )	25	W
$T_{op}$	Junction Operating Temperature	-25 to 130	$^\circ C$
$T_{stg}, T_j$	Storage and Junction Temperature	-40 to 150	$^\circ C$

**PIN CONNECTIONS (top view)**



**THERMAL DATA**

Symbol	Parameter		PowerSO20	Multiwatt15	Unit
$R_{th j-case}$	Thermal Resistance Junction-case	Max.	-	3	$^\circ C/W$
$R_{th j-amb}$	Thermal Resistance Junction-ambient	Max.	13 (*)	35	$^\circ C/W$

(\*) Mounted on aluminum substrate