

LXA  
250



**UNIVERSIDAD NACIONAL AUTONOMA  
DE MEXICO**

**FACULTAD DE ESTUDIOS PROFESIONALES  
CUAUTITLAN**



**DISEÑO DE UNA BALANZA ELECTRONICA CONTROLADA  
CON EL MICROPROCESADOR iAPX-8088**

**T E S I S**  
QUE PARA OBTENER EL TITULO DE:  
**INGENIERO MECANICO ELECTRICISTA**  
P R E S E N T A N :  
**JOSE FRANCISCO MARTINEZ AGUIRRE**  
**EDGARDO TAPIA CRUZ**

**Asesor: I.M.E. Blanca de la Peña Valencia**

**CUAUTITLAN IZCALLI, EDO. DE MEX.**

**TESIS CON  
FALLA DE ORIGEN**

1993



## **UNAM – Dirección General de Bibliotecas Tesis Digitales Restricciones de uso**

### **DERECHOS RESERVADOS © PROHIBIDA SU REPRODUCCIÓN TOTAL O PARCIAL**

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

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

# I N D I C E

INTRODUCCION	iv
CAPITULO I	
EXTENSOMETRIA : LA GALGA EXTENSOMETRICA, TIPOS Y CARACTERISTICAS	1
1.1 Fundamentos de la extensometria	2
1.2 Principio de medida de la galga extensometrica	4
1.3 Caracteristicas generales de las galgas	6
1.4 Criterios para la seleccion de una galga	10
1.5 Eleccion de la galga	12
1.6 Calculos para el diseno de la viga cantilever	14
1.7 El puente de Wheatstone	19
1.8 Caracteristicas del maquinado e instrumentacion del transductor	21
1.9 Pruebas de instrumentacion y funcionamiento	28
1.10 Conclusiones que arroja la observacion de las pruebas de instrumentacion	29
CAPITULO II	
CARACTERISTICAS GENERALES DEL MICROPROCESADOR 8088	31
2.1 Caracteristicas principales del 8088	36
2.2 Arquitectura tubular (pipeline)	38
2.3 Juego de registros	40
2.4 Senales y terminales	42
2.5 Estructura de interrupciones	46

2.6 Modos de direccionamiento	49
2.7 Estructura de memoria de segmentación	50

### CAPITULO III

3.1 Circuito básico para un sistema controlador basado en el microprocesador 8088	51
3.2 Circuito de reloj	53
3.3 Instalación del microprocesar (µP 8088)	60
3.4 Demultiplexión de buses	64
3.5 Instalación de memoria RAM y ROM	68
3.6 Puerto de entrada salida	73
3.7 Display	78
3.8 Teclado para el sistema controlador	83
3.9 Instalación del sistema de interrupciones	93
3.10 El controlador de interrupciones programable 8259A	96
3.11 Mapa de memoria del sistema y la ubicación en el mismo de los vectores de interrupción	111

### CAPITULO IV

4.1 Etapa de amplificación de la señal	120
4.2 Conversión analógica a digital. Convertidor de integración	128
4.3 Descripción del sistema de conversión	130
4.4 Descripción del procesador analógico	134
4.5 Unión del procesador analógico al sistema	137

4.6 Fuente de alimentación	142
<b>CAPITULO V</b>	
4.1 Programación	147
<b>CONCLUSIONES</b>	161
<b>APENDICE A</b>	
Tipos de galgas extensométricas y cuadro comparativo	163
<b>APENDICE B</b>	
Hojas de datos, diagramas de tiempos de circuitos integrados y juego de instrucciones del ensamblador 8088	167
<b>BIBLIOGRAFIA</b>	183

## I N T R U D U C C I O N

El desarrollo actual de los microprocesadores y sus lenguajes de programación permiten su utilización en equipos de medición que puedan ser adaptados mediante un cambio en su programación a una gran variedad de operaciones, donde no solo se mida un fenómeno físico sino que se que pueda dar un tratamiento a la información obtenida para incrementar su utilidad y poder satisfacer las necesidades que se tengan.

El primer capítulo comienza explicando que es la técnica de la extensometría, que son las galgas extensométricas así como sus tipos y características. Además se describen los pasos para instrumentar una viga cantilever como transductor. Cabe señalar que la viga cantilever fué instrumentada por lo cual se incluyen al final de este capítulo los resultados de las pruebas a que fué sometida.

El objetivo principal de la extensometría óhmica es el de aprovechar las propiedades electromecánicas de aleaciones conductoras para poder transformar un fenómeno mecánico en un respuesta eléctrica, así se desarrolla "la galga extensométrica", que es una resistencia especialmente diseñada para aprovechar las propiedades de algunas aleaciones de cobre (constantan) y otros materiales que varían su resistencia óhmica al ser deformados térmica ó mecánicamente. El control de calidad

para producir estos elementos transductores es tan preciso que puede confiarse que su variación óhmica es muy lineal respecto de su deformación mecánica, tanto que se producen en rangos de deformación-variación de resistencia de acuerdo a las elongaciones en materiales específicos.

Es así como pueden crearse dispositivos que censuren fielmente fenómenos como cambio de peso, a consecuencia de la variación de masa, que actúa deformando el elemento transductor. Al crearse elementos transductores de alta sensibilidad a la deformación mecánica, se pueden construir balanzas que no dependan de diseños en los que intervengan partes mecánicas móviles que están expuestas al desgaste.

La presente tesis hace uso de esta tecnología al instrumentar un transductor que aprovecha la deformación de una barra metálica de propiedades documentadas a la cual se le han adherido galgas extensométricas como detectores de deformación la cual es proporcional a las variaciones de peso que soporta la viga. La viga se diseñó de tal forma que no es un conjunto de formas que requieren de numerosos y complicados estudios y pruebas de mecánica para llegar a un modelo cuya sofisticación no es en sí, el objetivo de esta tesis. Se evitó la realización de tales estudios porque un diseño "novedoso" implica la realización de suficientes pruebas de esfuerzo, e incluso de instrumentación extensométrica, que equivale a hacer varias veces el transductor solo para averiguar el comportamiento

práctico de un complicado y especializado diseño teórico . Esto redunda en elevar el nivel del proyecto hasta convertirlo en una tesis de mecánica exclusivamente, pero también eleva los costos por utilización de equipo de laboratorio y materiales como probetas y extensímetros , lo cual no es un objetivo práctico para esta tesis.

Una forma confiable para el transductor , es una viga cantilever cuya forma y análisis de esfuerzos se puede determinar con fiabilidad con un mínimo de pruebas prácticas siendo su examen, precisamente , su funcionamiento como transductor.

En el segundo capítulo se ven las razones y ventajas de utilizar un microprocesador para el control de la balanza electrónica. A continuación se introducen las características básicas del microprocesador IAPX-8088 y se describe su estructura.

En el tercer capítulo se describen las partes que integran el sistema controlador basado en el microprocesador IAPX-8088 y se explican los pasos necesarios para integrar cada etapa , así como el porque de la utilización de estas.

En el siguiente capítulo se analizan otras etapas necesarias para el funcionamiento de la balanza electrónica: la etapa de amplificación de la señal , la etapa de conversión analógica a digital y la fuente de poder.

En el quinto capítulo se encuentra el programa utilizado por la balanza electrónica y se da una explicación de las rutinas que integran el programa.

## C A P I T U L O I

### **EXTENSOMETRIA: LA GALGA EXTENSOMETRICA, TIPOS Y CARACTERISTICAS.**

Esta técnica tiene por objeto la medida de las deformaciones unitarias experimentadas por los cuerpos sometidos a determinadas acciones exteriores y la obtención a través de ellas del estado superficial de tensiones existentes en cada punto. Esta técnica es insustituible, no existe procedimiento mecánico alguno que permita obtener ni aproximadamente este tipo de datos. Las limitaciones de su uso han sido debidas a las dificultades tecnológicas de su puesta en práctica. Pero el desarrollo experimentado de la electrónica en estos últimos años ha facilitado de tal modo su aplicación que ha hecho que la extensometría deje de estar al servicio exclusivo de los laboratorios y sea cada vez más utilizada en los procesos industriales de fabricación.

Numerosos posibles utilizadores comprenden las ventajas de esta técnica, pero su especialidad no es la electrónica, y se encuentran con dificultades no solo en la elección del equipo mejor adaptado a la resolución de su problema y en la utilización práctica de las galgas, sino hasta en el procesamiento de la información.

### 1.1.- FUNDAMENTOS DE LA EXTENSOMETRIA.

a) Definición: "La extensometría tiene por objeto la medida de las deformaciones superficiales de los cuerpos y por ello sirve para evaluar todas las magnitudes físicas capaces de originar deformaciones superficiales en los cuerpos sobre los que actuen"<sup>1</sup>.

b) Unidad de medida: La magnitud utilizada es adimensional y se denomina "deformación unitaria" =  $\Delta l/l$ . La unidad correspondiente es la "microdeformación" ( $\mu\epsilon$  = microstrain).

$$1 \mu\epsilon = 10^{-6} \text{ m/m} = 1 \mu\delta$$

c) Principio de medida: Un transductor llamado "galga"<sup>2</sup> se une sólidamente al cuerpo cuya deformación quiera medirse para que experimente sus mismas variaciones de longitud. La resistencia eléctrica de la galga es función de su longitud y su variación de longitud se obtiene midiendo la variación de su resistencia eléctrica.

Ecuación fundamental :  $\Delta R/R = k (\Delta l/l)$

Siendo R = resistencia eléctrica de la galga.

$\Delta R$  = variación de resistencia.

L = longitud de la galga.

$\Delta l$  = variación de longitud.

k = factor de galga.

1. APUNTES DE EXTENSOMETRIA

2. EL TRANSDUCTOR LLAMADO GALGA TAMBIEN ES CONOCIDO COMO EXTENSIMETRO OHMICO, GALGA EXTENSOMETRICA O STRAIN GAGE

d) Métodos de medida:

d.1.- Directo : Cuando lo que se mide es la variación de resistencia que experimenta al variar su longitud. La precisión es grande pero sólo se efectúan mediciones en régimen estático.

d.2.- Indirecto : Cuando no se mide directamente la variación de resistencia, sino una magnitud que es función de ella, como la variación de voltaje ó intensidad. Por introducir más conversiones de magnitudes, la precisión de este método es menor pero se pueden realizar mediciones en régimen dinámico. La característica dinámica de esta forma de medición se aprovecha en esta tesis para un muestreo continuo del fenómeno a medir.

e) Desarrollo histórico: El estudio de la variación de la resistencia en función de la variación de sus características mecánicas fué iniciado en 1856 por Thomson .

En el año 1935, el profesor Simmons, de U.S.A., realizó las primeras experiencias sobre hilos metálicos encolados y en el año de 1937 el profesor Ruge creó las primeras galgas constituidas por un hilo metálico plegado varias veces sobre sí mismo.

Al entrar en guerra los Estados Unidos se utilizó de manera sistemática la extensometría en estudios de construcciones navales y aeronáuticas.

f) Dominio de utilización: Las deformaciones unitarias medibles dependen de la galga extensométrica utilizada. Desde 50 hasta 50,000  $\mu\epsilon$  son valores alcanzados normalmente, pero en condiciones especiales pueden medirse desde 1 hasta 200.000  $\mu\epsilon$ .

Las galgas son muy sensibles a las variaciones de temperatura y hay que proveer montajes adecuados para su compensación. Desde  $-200^{\circ}\text{C}$  hasta  $+180^{\circ}\text{C}$  es la zona normal de trabajo pero con galgas de base de asbesto se pueden alcanzar los  $400^{\circ}\text{C}$  y con galgas de base metálica se puede llegar hasta los  $1100^{\circ}\text{C}$ .

La longitud de la zona del cuerpo cuya deformación se mide puede variar desde 0.3mm. hasta 150 mm. en aplicaciones muy especiales.

Los factores ambientales tales como presión, humedad, aceite, ácidos corrosivos, abrasivos, campos eléctricos y magnéticos, etc., no limitan el dominio de la extensometría si se cuida adecuadamente la elección de la galga y la protección adecuada.

## 1.2.- PRINCIPIO DE MEDIDA DE LA GALGA EXTENSOMETRICA.

“La variación de la resistencia eléctrica de la galga es proporcional a la variación de su longitud”.

La galga clásica está constituida por un hilo metálico cilíndrico replegado varias veces sobre sí mismo como se indica en la figura siguiente:

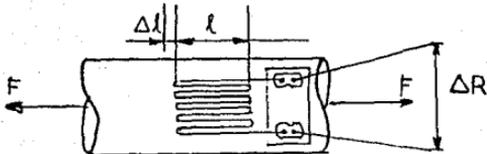


Fig. 1.1

Sea:

$L$  = longitud inicial de la galga

$R$  = resistencia inicial de la galga

$S$  = sección circular del hilo metálico

$r$  = radio de la sección circular

$\rho$  = resistividad del hilo metálico

$\nu$  = coeficiente de Poisson del hilo metálico

$$R = \rho(L/S) \quad (1)$$

$$R = \rho(L/S) = \rho(L/(\pi r^2)) \quad (2)$$

diferenciando:

$$dR = d\rho(L/(\pi r^2)) + \rho(L/(\pi r^2))dL - \rho L(2dr/(\pi r^3)) \quad (3)$$

dividiendo por R:

$$dR/R = d\rho/\rho + dL/L - 2dr/r \quad (4)$$

y como

$$dr/r = -\nu(dL/L) \quad (5)$$

obtenemos

$$dR/R = dp/\rho + dL/L + 2\nu(dL/L)$$

$$dR/R = dp/\rho + (dL/L)(1+2\nu) \quad (6)$$

Se ha constatado experimentalmente que:

$$dp/\rho = c(d\nu/\nu) \quad (7)$$

y como

$$\nu = L/rE^2 \quad (8)$$

diferenciando

$$d\nu/\nu = dL/L + 2dr/r = (1-2\nu)(dL/L) \quad (9)$$

sustituyendo en (7)

$$dp/\rho = c(1-2\nu)(dL/L) \quad (10)$$

sustituyendo en (6), obtenemos

$$dR/R = (dL/L)(1+2\nu+c(1-2\nu)) \quad (11)$$

y llamando factor de galga a la cantidad  $K = 1+2\nu+c(1-2\nu)$

obtenemos la ecuación fundamental de la extensometría:

$$dR/R = K(dL/L)$$

### 1.3.- CARACTERISTICAS GENERALES DE LAS GALGAS.

Las características generales más importantes de las galgas extensométricas se describen a continuación y diversos tipos de ellas y un cuadro comparativo se muestra en el apéndice "A" al final de la tesis.

a) Resistencia de la galga.

Valores comerciales desde 120 ohm hasta 1000 ohm.

b) Factor de galga.

El factor de galga K se define como la variación de su resistencia entre su alargamiento unitario. Su valor nominal es 2, aunque puede alcanzar mayores dependiendo del material usado para su construcción. Los valores reales de sensibilidad se obtienen por muestreo de las distintas series de fabricación mediante ensayos de flexión. Estos pueden representar un error normalizado de hasta un  $\pm 2\%$  del valor nominal dado por el fabricante. Para las galgas normales su valor es 2 y para las semiconductoras llega hasta 200.

c) Histéresis mecánica y linealidad.

Si representamos el diagrama resistencia-deformación de la galga cuando cargamos ésta después de su pegado, ocurre en primer lugar, que la galga se deforma linealmente hasta un valor (1) a partir del cual no se cumple la relación.

$$\Delta R/R = K$$

Descargada la galga (Fig. 1.2) queda una deformación permanente  $\epsilon_0$  debida a la histéresis de los materiales usados. Cargando y descargando posteriormente la galga se observa que su comportamiento no es el estrictamente lineal, sino del tipo representado en la Fig. 1.2. La linealidad se representa en forma de desviación máxima respecto a la línea teórica 0-1 para  $1.000 \mu\epsilon$ .

Para evitar estos errores es aconsejable aplicaciones repetidas de carga antes de la medida definitiva.

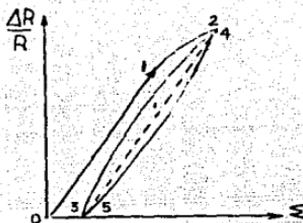


Fig. 1.2

d) Límite de fatiga.

Viene expresado por el número de ciclos que aguanta a un esfuerzo determinado.

e) Sensibilidad transversal.

Acompañando a la deformación que sufre la galga (sometida a una sollicitación cualquiera) en el sentido longitudinal ó principal de medida se produce otra secundaria en el sentido transversal, que en determinados casos puede alterar notablemente la exactitud de la medida. En el caso de una galga (cargada) alineada perfectamente con la dirección de la deformación principal, aparece otra deformación (de signo contrario por la ley de POISSON) en sentido transversal que será proporcional a la deformación principal y a la sensibilidad en el sentido transversal. Interesa por consiguiente, reducir al máximo esta sensibilidad. En las galgas impresas se logra hacer prácticamente despreciable éste efecto engrosando el conductor en sus puntos de inflexión (meandros). La sensibilidad transversal se mide en % de la sensibilidad longitudinal. Un valor normal es el 1% de la sensibilidad longitudinal.

#### 1) Efecto de la temperatura.

Es muy importante el efecto de la temperatura ya que las deformaciones medidas en una galga no sometida a esfuerzos pueden alcanzar por efecto de la temperatura hasta 50 y 60  $\mu\delta$  como se indica en la figura 1.3.

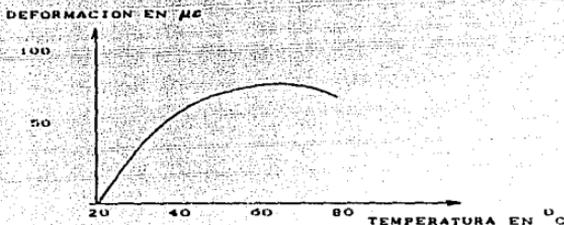


Fig. 1.3

Por ello es muy importante utilizar cuando es posible galgas autocompensadas ó en su defecto prever los dispositivos de montaje necesarios para lograr dicha compensación. Eliminan positivamente el efecto nocivo de los cambios de temperatura.

#### g) Duración de las galgas.

Practicamente ilimitada, siempre que se hayan cumplido correctamente los requisitos de pegado, aislamiento, condiciones ambientales y deformaciones máximas que se aconsejen para cada tipo.

h) Capacidad de trabajo.

La capacidad de carga estática de una galga está limitada, junto con otros factores, por su capacidad de deformación. Para galgas impresas pueden alcanzarse deformaciones de hasta 3-4% de su longitud activa, permitiéndose valores máximos sin error (depende del fabricante) de  $\pm$  4000 ó 5.000  $\mu$ ó. El comportamiento dinámico de las galgas es más difícil de predecir determinándose normalmente mediante ensayos repetidos.

#### 1.4.- CRITERIOS PARA SELECCION DE UNA GALGA

Conocidas las propiedades y características básicas de las galgas extensométricas, existen criterios elementales para la selección de estas en función de las condiciones del fenómeno que queremos medir.

a).Temperatura de utilización.

Es uno de los condicionantes fundamentales en la elección. Deben considerarse el siguiente aspecto:

-Material del conductor

La aleación más utilizada para temperaturas medias (-20°C a +60°C) es el constantan. Para temperaturas elevadas (hasta 300°C) son óptimas las aleaciones cromo-níquel. Para temperaturas muy elevadas se aconsejan galgas sin soporte o encapsulados cerámicos.

La utilización de galgas autocompensadas ó galgas adicionales de compensación (dummy) dependerá de cada situación en concreto (experiencia, economía, complejidad y condiciones del ensayo).

b).Tamaño de la galga.

Una de las características más importantes en la elección de la galga es su longitud activa (l). Es la distancia existente, longitudinalmente, entre los puntos de cambio de sentido (meandros) de los hilos conductores.

La galga promedia el esfuerzo existente dentro de su área activa. Por consiguiente si estos esfuerzos son uniformes, longitudes por encima de 5 ó 6mm. son aceptables. Si existe concentración de esfuerzos cuanto menor sea l mayor fidelidad se obtendrá.

En algunos transductores se emplean longitudes activas muy cortas y anchos grandes, pero se recomienda en general longitudes grandes.

Un segundo punto a considerar es el tipo de sollicitación (dinámica) de que se trate. En ensayos dinámicos de alta frecuencia existe una limitación. Como las deformaciones se propagan a la velocidad del sonido, una onda frontal de longitud igual a la de la galga no dará señal, ya que una mitad de la galga está alargada y otra contraída al mismo tiempo. Para evitar el problema se recomiendan longitudes activas iguales ó inferiores a  $l/6$  de la longitud de la onda del fenómeno a medir.

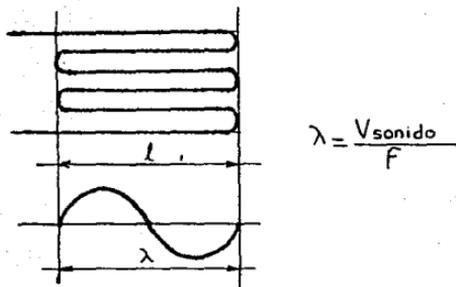


Fig. 1.4

c). Rango de deformación.

Si las deformaciones reales que han de producirse en el ensayo son muy pequeñas se deben utilizar las galgas semiconductoras.

Para deformaciones normales (un 2% de  $l$ ) existe una amplia gama en el mercado.

Para deformaciones elevadas (10-15%) es necesario emplear galgas especiales

d). Dirección de los esfuerzos.

Si se conoce la dirección de los esfuerzos a medir es suficiente la colocación de una galga en esa dirección. Si no se conoce ésta, se hace necesario el empleo de galgas multiaxiales, de distintos tipos, según se requiera.

### 1.5.- ELECCION DE LA GALGA

En base a los criterios anteriormente expuestos se requiere de una galga con las siguientes características :

a) Temperatura de utilización.

Para temperatura media (-20 a 60°C) y más específicamente temperatura ambiente, la aleación de constantan es adecuada y común en el mercado especializado. Además es posible utilizar galgas autocompensadas para completar un puente de Wheatstone 4/4 y hacer despreciables las variaciones por temperatura.

b) Tamaño de la galga.

Se aconsejan longitudes inferiores a 1/6 de la longitud de onda ( $\lambda$ ) del fenómeno a medir, así:

$$\lambda = v_{\text{sonido}}/f = (300\text{m/s})/(15 \text{ Hz}) = 20\text{m/ciclo}$$

15 Hz es la máxima velocidad de conversión del convertidor A/D.

c) Rango de deformación.

La deformación de la galga es del tipo normal ( $\Delta l = 0.02 L$ ) es muy pequeña.

d) Dirección del esfuerzo.

Esto depende del elemento que constituye el cuerpo transductor. En este caso una viga empotrada en cantilever donde las galgas medirán las deformaciones a flexión y a compresión cuando ésta es cargada en su extremo con determinados pesos. En este tipo de vigas se conoce el eje que es paralelo al esfuerzo generado.

Las galgas seleccionadas son las de film de hilo de constantan.

## 1.6.- CALCULOS PARA EL DISEÑO DE LA BARRA CANTILEVER.

El elemento que constituye el cuerpo transductor tiene la forma de una viga empotrada en cantilever y su material es una aleación de aluminio (Duraluminio 2024T-21).

Las ecuaciones utilizadas dan el cálculo del momento flexionante en una viga empotrada con las siguientes consideraciones de simplificación :

Consideraciones de simplificación del presente cálculo.

1. Los ejes X-Y son ejes principales de la sección recta y preferente el eje Y es un eje de simetría. Figura 1.5.
2. El plano YZ es un plano de simetría en el que actúan cargas, reacciones, fuerza cortante, momento flexionante y las resultantes de los esfuerzos internos y en él se efectúa la flexión.
3. La sección recta aa plana y normal a las fibras longitudinales Z antes de la deformación, se conserva plana y normal a las fibras longitudinales después de la deformación.
4. Durante la deformación la sección recta aa sufre un desplazamiento lineal y una rotación alrededor del eje X.
5. Dos secciones rectas paralelas se vuelven convergentes después de la deformación; si el momento flexionante que actúa en la viga es negativo, la convergencia es hacia abajo y la cara inferior de la viga se vuelve cóncava y

la cara superior convexa; las dos secciones rectas se aproximan en la parte inferior y se alejan en la parte superior.

6. Las fibras longitudinales situadas del eje X hacia abajo, soportan en consecuencia esfuerzo compresivo y las que están situadas del eje X hacia arriba, soportan esfuerzos por tensión y tienen un valor constante en los puntos situados a la misma distancia del eje X.
7. Las fibras longitudinales del plano XZ no sufren esfuerzos por compresión o tensión y por lo tanto ese plano recibe el nombre de plano neutro.
8. El material es homogéneo y obedece a la ley de Hooke.
9. El módulo elástico es igual a tensión que a compresión.
10. La viga es inicialmente recta y de sección constante.

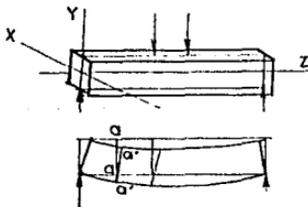


Fig. 15

Para el diseño de la viga es necesario conocer las variaciones que sufren tanto la fuerza cortante como el esfuerzo cortante en toda su longitud, lo cual se obtiene mediante el establecimiento de las ecuaciones generales y su representación en diagramas.

Para una viga empotrada en uno de sus extremos y con una carga concentrada en el extremo libre (figura 1.6) el equilibrio se establece con una reacción  $R = F$  y con un momento de empotramiento:

$$M = - F * L$$

En una sección cualquiera  $z$  :

$$V_z = -F \quad \text{y} \quad M_z = -Fz$$

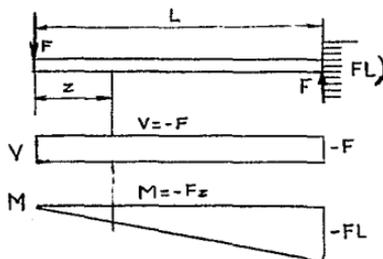


Figura 1.6

El esfuerzo  $\sigma_z$  es directamente proporcional al momento flexionante M y a la distancia y al eje neutro e inversamente proporcional al momento de inercia  $I_x$ . La ecuación del esfuerzo  $\sigma_z$  es:

$$\sigma_z = (M y) / I_x$$

Los esfuerzos normales son máximos en las fibras más alejadas del eje neutro, donde y es máxima y son nulos en el eje neutro.

$$\sigma_{z \text{ max}} = (M y_{\text{max}}) / I_x$$

El cociente  $S_x = I_x / y_{\text{max}}$  se llama módulo de la sección y se mide en  $\text{cm}^3$ .

Si se supone que un momento flexionante M actúa sobre una sección rectangular, el módulo de sección tendrá el valor siguiente:

$$I_x = (b h^3) / 12 \quad y_{\text{max}} = h / 2$$

$$S_x = \frac{1}{6} (b h^2)$$

Para el diseño de la viga se requieren los valores del momento flexionante máximo M que actúa en la viga y el esfuerzo admisible máximo que debe aplicarse al material de que está hecha la viga. Determinando el módulo de la sección  $S_x$  es posible diseñar la sección y darle las dimensiones necesarias.

$$S_x = M / \sigma$$

La aleación de duraluminio tiene un esfuerzo de trabajo

$$\sigma = 3800 \text{ kg/cm}^2.$$

La carga que se va a aplicar a la viga será de 200gms. pero por razones de seguridad se hacen los cálculos para una carga de 500gms.

La distancia del punto de aplicación de la carga al eje de las galgas es de 16.5 cm.

$$\sigma = 3800 \text{ kg/cm}^2$$

$$F = .5 \text{ kg}$$

$$l = 16.5 \text{ cm}$$

$$b = 1.9 \text{ cm}$$

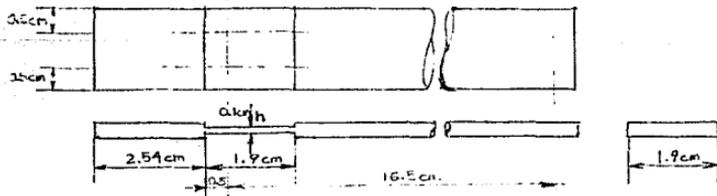
$$M = F(d) = .5 \text{ kg} (16.5 \text{ cm}) = 8.25 \text{ kg-cm}$$

$$S_x = M / \sigma = (8.25 \text{ kg-cm}) / 3800 \text{ kg/cm}^2 = 0.00217 \text{ cm}^3$$

$$h^2 = (6 S_x) / b = (6 * 0.00217 \text{ cm}^3) / 1.9 \text{ cm} = 0.00685 \text{ cm}^2$$

$$h_{\text{mínima}} = 0.0827 \text{ cm}$$

Se elige  $h = 0.1 \text{ cm}$  porque es mayor que la  $h$  mínima.



Dimensiones de la viga cantilever.

Figura 1.7

### 1.7.- EL PUENTE DE WHEATSTONE.

Como se vio, el efecto de deformación de la galga se convertía en una variación de su propia resistencia eléctrica; para medir dicha variación se utiliza el puente de Wheatstone. El puente de Wheatstone está formado por cuatro impedancias montadas dos a dos, en serie, que están alimentadas por una fuente de voltaje  $V_e$  (figura 1.8) con una resistencia que se supone nula. Sea  $G$  un aparato de medición cuya resistencia se supondrá infinita, frente a los valores de  $r_1$ ,  $r_2$ ,  $r_3$  y  $r_4$ . Suponiendo que el puente no está equilibrado y que tiene un voltaje  $V_s$ , entonces :

$$V_s = V_e (r_1 / (r_1 + r_3) - r_2 / (r_2 + r_4))$$

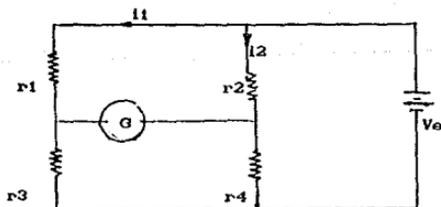


Fig. 1.8

Si se hacen variar la resistencia de las galgas en los valores  $\Delta r_1$ ,  $\Delta r_2$ ,  $\Delta r_3$  y  $\Delta r_4$  la variación del voltaje de salida será:

$$\Delta V_s = V_e \left[ \frac{r_1 + r_3}{(r_1 + r_3)^2} * \frac{\Delta r_1}{r_1} - \frac{\Delta r_3}{r_3} \right] - \left[ \frac{r_2 + r_4}{(r_2 + r_4)^2} * \left( \frac{\Delta r_2}{r_2} - \frac{\Delta r_4}{r_4} \right) \right]$$

Para el caso de que el puente estuviera equilibrado  $V_s=0$  lo que exige que  $r_1/r_3 = r_2/r_4$  y si  $r_1=r_2=r_3=r_4$  entonces:

$$\Delta V_s = (V_e/4) * (\Delta r_1/r_1 - \Delta r_2/r_2 - \Delta r_3/r_3 + \Delta r_4/r_4)$$

De la ecuación anterior se deduce la regla de los signos que se puede enunciar de la forma siguiente:

Dos ramas opuestas influyen la tensión de salida en el mismo sentido es decir las ramas 1 y 4 en sentido opuesto a las ramas 2 y 3.

Para la viga cantilever que se utiliza en esta tesis las galgas 1 y 4 deberán ser colocadas sobre la cara sometida a tensión y las galgas 2 y 3 en el lado que trabaja a compresión con el fin de obtener un aumento de la señal de salida.

1.8- CARACTERISTICAS DEL MAQUINADO E INSTRUMENTACION  
DEL TRANSDUCTOR.

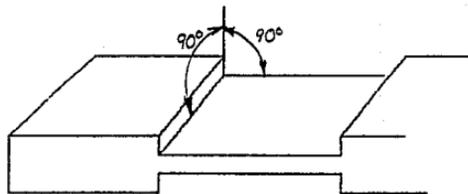
1.-FIG.1.9a : El maquinado debe ser lo más recto posible para evitar el corte transversal sobre la barra a su eje longitudinal, para no desalinear los ejes de las galgas. Este mismo corte debe ser a escuadra para que se aproveche al máximo el momento y que las galgas estén lo más cerca al empotramiento "perfecto" .

2.-FIG.1.9b : Para una pieza que mide esfuerzos en una dirección es importante un solo eje.

Como solo se utiliza un eje de las galgas, el error del corte escuadra puede ser tolerado al poner la base de las galgas un poco alejadas ó promediando un acercamiento a la pared que esta más derecha para evitar montar en desnivel algunas de las bases de las galgas.

En caso de requerir del estudio de más de un eje con galgas unidireccionales, debe vigilarse la alineación, con gran precisión, de los ejes que se deseen estudiar: (a 45o,90o y configuraciones especiales) .

3.-FIG.1.9c : Para el caso de la barra hay que trazar las direcciones a estudiar todas perpendiculares por parejas (un eje " x " con su respectivo " y " y así sucesivamente). Los ejes de las galgas deben alinearse para el estudio de esfuerzo en las mismas direcciones de estos.

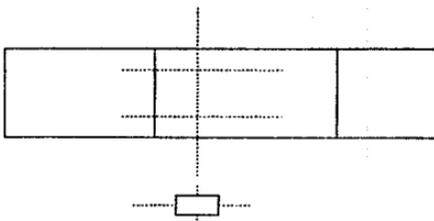


MAQUINADO

FIGURA 1.8A

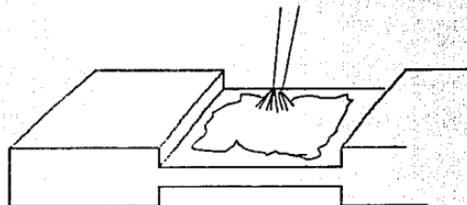


FIGURA 1.8B



SE TRAZAN LOS EJES DE REFERENCIA

FIGURA 1.8C



SE APLICA EL ACONDICIONADOR  
Y POSTERIORMENTE EL NEUTRALIZADOR

FIGURA 1.8D

El trazo de los ejes debe hacerse (en la barra) con lápiz muy duro como el que se utiliza para rayar lámina de acero. Después del trazado (muy suave) se le trata con ácidos para crear una superficie en relieve como el trabajo en placas con cubierta de cobre para la fabricación de circuitos impresos. No deben existir en esta superficie rayones o marcas de cualquier tipo que afecten las capas internas del material y el comportamiento elástico de las mismas.

Lo que se procede a hacer es conjuntar todo el material necesario para la instrumentación:

- Galgas aptas para el material (de propósito general ó especiales para el módulo de elasticidad particular y bien determinado " E ").
- Pegamento (cemento de resina que requiere de horneado).
- Limpiador y acondicionador.
- Tijeras para cortar la base de las Gages o cualquier implemento que así lo requiera.
- Pinzas de electrónica y de joyero.
- Gasas estériles
- Plácas de goma y aluminio para la distribución de la presión cuando se pegan las galgas al material.
- Lente de joyero.
- Lupa.
- Escudetas o charolas de trabajo donde se viertan excedentes de líquidos o se reciban piezas muy pequeñas o difíciles de manipular y que no se puedan tocar con las manos.

#### PREPARACION DE LA SUPERFICIE

Se procede a realizar los siguientes pasos después de examinar que la pieza no esté maltratada en la zona de instrumentación :

4.-FIG.1.9d : Desengrasado de la superficie en una sola dirección con una aplicación del ACONDICIONADOR que en realidad es ACIDO FOSFORICO en una solución muy suave aplicada con un pincel o por goteo. Se tratarán con el acondicionador ambas caras procurando no tocar con los dedos ó el procedimiento se repite de nuevo.

5.-FIG.1.9e : Se limpia con un pedazo de gasa estéril en una sola dirección de tallado. Después de la primera pasada con gasa húmeda, se limpia con gasa seca para evitar la erosión de la superficie.

Se procede a neutralizar la acción del acondicionador con un neutralizador que en realidad es AMONIACO, también en una solución muy suave, con lo que la pieza queda lista para el proceso de instrumentación.

Las conexiones no se deben hacer en las platinas de las galgas en su totalidad, lo más seguro es que se quemem mucho antes de verificar su correcta conexión. Para llevar a cabo este paso se requieren de pequeñas placas de conexión hechas de mica y el material conductor es una película de cobre. Estas placas sirven de conexión con las galgas y el equipo externo. Las galgas reciben calor de soldadura solo una vez, cuando se

conecta el alambre que las une a la placa de conexión que es donde todas las maniobras de alambrado se realizan (instrumentación de puentes de Wheastone u otras configuraciones, unión con resistencias de compensación, etc.) y de donde parten los cables que conectan al transductor con el equipo de proceso de datos. Estas placas son mucho más robustas que las galgas y corresponde una para cada galga utilizada en el transductor.

6.-FIG.1.9f : Se procede al pegado y alineación preliminar es entonces cuando se vigila la alineación y el centrado. Como se desea medir el esfuerzo en una dirección las galgas podrian ir pegadas a una de las orillas ó centradas. para este caso se prefirieron centradas. Las galgas se sitúan sobre la superficie de aplicación cuidando que sus ejes se encuentren alineados con las referencias tomadas y junto a ellas se sitúan las placas de conexión. Todo con ayuda del lente de joyero. Se procede a mantener el conjunto (galga y placa de conexión respectiva) unido con una cinta adhesiva que resista las temperaturas requeridas para el horneado, cuidando de dejar sin pegar un extremo de la cinta para facilitar levantarla cuando se requiera.

7.-FIG.1.9g : Después de crear dos grupos de galgas y placas de conexiones unidas con cinta, se levanta esta dejando adherido un extremo.

8.-FIG.1.9h : Se procede al uso del pegamento. Este es un cemento cuya propiedad es endurecerse mediante el horneado a aproximadamente 400°C. Una vez horneado el cemento no se espere poder despegar ni una partícula de las galgas.

9.-FIG.1.9i : Ahora sigue el proceso de inspección con lente de joyero donde se observa si las galgas se han arrugado (puede corregirse solo de manera inmediata y en movimientos muy precisos) o si existen burbujas en el cemento, en este caso se levantan las galgas y se hace uniforme la capa de pegamento.

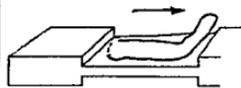
10.-FIG.1.9j : Se procede a proteger el conjunto instrumentado con placas de goma y aluminio, puestas en ese orden de contacto con la superficie de las galgas, con estas placas distribuye la presión ejercida por las pinzas cargadas a resorte que se utilizan para mantener en su lugar a las galgas durante el horneado.

El horneado se lleva a cabo durante 2 horas dejando enfriar de manera natural el horno sin abrirlo.

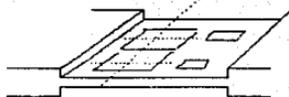
Se procede a quitar cuidadosamente las pinzas, las placas de aluminio y finalmente las de goma para una inspección de la barra.

Para las conexiones de instrumentación se utiliza un cautín de calor controlado, para que la punta este a aproximadamente 400°C (Ver la figura 1.9k de conexiones)

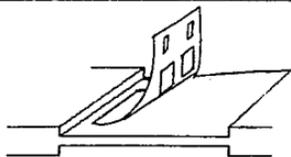
Al terminar de soldar, se aplica un desengrasante para eliminar los restos tanto de la resina de la soldadura como del pegamento de la cinta adhesiva usadas con anterioridad.



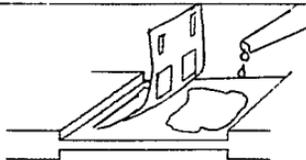
SE LIMPIA CON GASA  
EN UNA DIRECCION  
FIGURA 1.9C



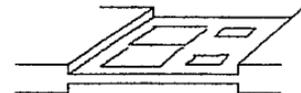
SE ALINEAN LAS CALZAS  
FIGURA 1.9F



SE DEJA PEGADO UN EXTREMO  
DE LA CINTA  
FIGURA 1.9G



SE APLICA EL PEGAMENTO  
FIGURA 1.9H



SE REVISIA EL PEGADO  
FIGURA 1.9I

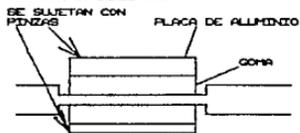
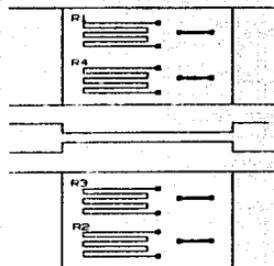


FIGURA 1.9J



R1 Y R4 SUJETAS A TENSION

R2 Y R3 SUJETAS A COMPRESION

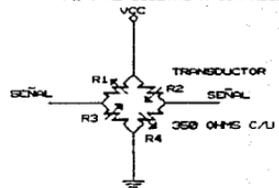


FIGURA 1.9K

## 1.9.- PRUEBAS DE INSTRUMENTACION Y FUNCIONAMIENTO.

### PRUEBA DE DERIVA A TIERRA.

Con un Megger se procede a medir, desde las conexiones al exterior. Si existe derivación a tierra o cortos entre los elementos transductores y la barra o a partir de las placas de base de las galgas a las mismas y la barra. La resistencia que deben exhibir debe ser mayor a los 20 M $\Omega$ . Sino es así, el proceso seguido desde el principio hasta el momento debe repetirse.

### PRUEBA COMO TRANSDUCTOR.

A continuación se monta la barra, tal como va a ser usada para transformar flexiones de esta, debidas al peso, en señal eléctrica y se hacen las conexiones con equipo para puente de Wheastone de instrumentación.

La aprueba arroja los siguientes datos:

ALIMENTACION DE 10V (Máximo )

UNIDADES:	PESO	(Gramos )
	SERIAL	( $\mu\epsilon$ microstrains, micro deformaciones)

Para:	100grs	3781 $\mu\epsilon$
	130grs	4915 $\mu\epsilon$
	200grs	7561 $\mu\epsilon$
	0grs	0 $\mu\epsilon$

Sensibilidad de entrada = 3.78 mV/VALIM

Salida a escala completa = 37.8mV para 200grs.

1.10. CONCLUSIONES QUE ARROJA LA OBSERVACION DE  
LAS PRUEBAS DE INSTRUMENTACION

1.- Se requiere de un filtro pasabajas ya que detecta la vibración de la barra al colocar un peso.

2.- Se logra buena respuesta inmediata, comprobando que el empleo de un transductor cuya construcción era más elaborada y complicada, es más redituable que el uso de una sola galga para crear un transductor que presenta problemas de estabilidad y sensibilidad debido a su sencillez.

3.- Se experimentó que si soportó 200 gramos y en el medidor de esfuerzos la lectura regresó a 0  $\mu$ c, lo cual significa que la barra trabaja en la zona elástica. NO HAY DEFORMACION PERMANENTE.

4.- Se sugiere el empleo de una referencia que compense el peso de la barra y plato de la balanza.

5.- Es muy importante adecuar un tope para que la deformación no exceda la permitida para 200grs. No obstante la barra fué diseñada para operar a un peso máximo de 450grs.

6.- La última placa plástica de conexiones debe pegarse en el soporte con el mejor pegamento a disposición y además dejar los alambres de conexión (los que van de la placa que comunica con el equipo digital y las galgas directamente) lo suficientemente largos para que no exista restricción en el movimiento de la barra.

7.- Cuando se haya instrumentado por completo el transductor, debe ser impermeabilizada la superficie de las galgas con algun barniz que guarde la misma capacidad de deformación de éstas y del material del que esta hecha la barra. Como última observación, debe evitarse que la barra sea expuesta al maquinado de cualquier tipo debido a que el impacto de rebabas de material sobre las galgas aun protegidas daña irremediabilmente a las mismas.

## C A P I T U L O   I I

### CARACTERISTICAS GENERALES DEL MICROPROCESADOR 8088

La función principal de una balanza es mostrar mediciones de peso de una muestra. La balanza debe poseer características bien definidas , como son la precisión , la confiabilidad del aparato (si se piensa en un mercado potencial ésta la convierte en un producto "competitivo"), un rango de trabajo específico (determinando así su área de utilización ) ó bien la sensibilidad que es afectada o regulada por los anteriores factores pero que a su vez respalda a los mismos , especialmente cuando la exactitud de las pesadas es muy crítica Si el peso a medir es mucho muy pequeño se deben satisfacer necesidades de trabajo con mediciones de peso exacto y para obtenerlo es primordial el uso de equipo muy sensible (cuyo rango de medición, por razones técnicas se estrecha).

Si un equipo cumple con las cualidades antes mencionadas puede ser utilizado con fiabilidad para el fin para el que fué diseñado. No obstante, es cada vez mayor la necesidad de instrumentos que no solo sean confiables sino que ofrezcan facilidades que mejoren el aprovechamiento de los mismos , como la determinación por el usuario de unidades de medida ú otras operaciones que la persona ya no tenga que realizar de manera repetitiva. En muchas áreas de la industria ó en los laboratorios se requiere de muestreo estadístico ú operaciones técnicas , como conversión de unidades , entre tantas más que se

pueden citar.

Para lograr esto es necesario crear un equipo cuya característica distintiva sea la flexibilidad y esta se consigue sólo con la utilización de 'equipo auxiliar que pueda reconfigurarse dentro de ciertos rangos es decir, que se pueda adecuar su funcionamiento según las exigencias presentes. Esto solo es posible con la utilización de equipo que cambie su funcionamiento mediante ordenes recibidas con sistemas basados en un microprocesador ( $\mu P$ ) o un microcontrolador ( $\mu C$ ).

¿Qué diferencia a un microprocesador ó a un microcontrolador de equipo diseñado para ejecutar una función específica?

La diferencia radica precisamente . en que esta dentro de sus funciones la posibilidad de cambiar su comportamiento debido a órdenes que le son programadas y así no realizar una sola tarea sino, reiterando, que tiene la capacidad de adaptarse a distintas operaciones .

Un  $\mu P$  o  $\mu C$  puede adaptarse mediante programación a infinidad de operaciones , desde realizar una conversión A/D ,que es el principio básico de este proyecto , hasta controlar un proceso o secciones del mismo.

La balanza diseñad., en esta tesis puede controlar circuitos de función específica como convertidores A/D , lógica para controlar una pantalla , etc., pero para poder ofrecer algo más que una conversión de señal eléctrica debida a un peso

determinado y mandarlo a la pantalla , es necesario no solo el control del sistema abocado a la conversión A/D sino que se ejecuten operaciones de control de pantalla y lo más importante, manejo de datos, no solo el valor de una conversión deformación mecánica a señal eléctrica y lectura de peso sino además el tratamiento de estos datos , la emisión de mensajes útiles al usuario o la ejecución de programas que incrementen la utilidad de las mediciones.

Lo antes mencionado no se logra sin un sistema que CONTROLE el funcionamiento del equipo que conforma la balanza como sistema básico , PROCESA los datos obtenidos y los TRANSFIERA a donde sean útiles (un almacenamiento o mostrarlos en un exhibidor).

El ejecutor de todos estos pasos puede ser un  $\mu P$  pero éste a su vez necesita de donde obtener las instrucciones, una memoria permanente o temporal y donde almacenar cantidades cuyo uso sólo sea momentáneo (usando memoria temporal).

Así , es natural pensar en la utilización de un sistema ejecutor (un  $\mu P$ ) , donde almacenar los programas que describen las funciones que se desea realice la balanza (memoria ROM) , donde tenga los datos temporales (memoria RAM) , equipo para comunicar este sistema con el "exterior" , ya sea la etapa de conversión A/D , la pantalla , o un teclado a otros equipos distintos a la balanza (lógica I/O). Contando además con equipos de apoyo , comunicación del  $\mu P$  con todos los circuitos mencionados y lógica para interfazar el  $\mu P$  y sus periféricos , etc.

Cuando todo esto se juzga necesario utilizar se esta conjuntando un sistema completo de microproceso y en el presente caso un sistema básico.

#### ¿Porqué la elección del 8088?

A pesar de que se ha hablado ya de este tema ha llegado el punto donde se vuelve a revisar las razones de utilización de un  $\mu P$  que brinda las características necesarias para controlar no solo la balanza o cualquier otro tipo de proceso sino hasta computadoras de alto rendimiento.

#### Versatilidad

Dentro del mundo de los microprocesadores el 8088 destaca no sólo como un circuito de gran capacidad de manejo de datos (puede direccionar hasta un megabyte de memoria ). Contiene u obedece un ingenioso sistema de acceso a este tamaño de memoria siendo contenido en un DIP de 40 terminales.

El 8088 es el puente entre los microprocesadores de 8 bits y la nueva generación de  $\mu P$  de 16 bits , utilizando mucho del hardware de soporte para equipos más antiguos (como Z80 , MC6800 ú 8085),siendo un  $\mu P$  de 16 bits. Tiene un esquema de transferencia de datos " utiliza un poderoso esquema de interrupciones , no utilizadas en equipos anteriores a este.

Su bus de datos "restringido" de 8 bits le permite comunicarse con mucho equipo existente en el mercado (pantallas alfanuméricas, de matriz de puntos, controladores de 8 bits, sistemas de entrada-salida, memorias con bus de datos de 8 bits, etc).

#### Modernización de métodos y tecnologías .

La experiencia anterior de los expositores con equipo Zilog Z80 hace pensar en la posibilidad de utilizar este  $\mu P$ . Las cualidades de este equipo y sus sistemas de soporte lo hacen idóneo para realizar las tareas deseadas en esta tesis. Más continuar usandolo limita las posibilidades de aprendizaje y adaptación a nuevos equipos así como a software o elementos accesorios nuevos. Lo nuevo y moderno, no desplaza a lo antiguo cuando esté aun es vigente (el Z80 sigue siendo utilizado, en la actualidad cuesta aproximadamente una quinta parte del precio del 8088!). Pero es bueno tomar el hardware y software relacionado al Z80 como base para la comprensión de la tecnología de procesadores de 16 bits y su funcionamiento ; eventualmente se tiene que hacer lo mismo con el 8088, ya que en la actualidad existe una gran demanda por controladores que usen el 80188 (cuyo software se sitúa dentro de la misma familia del 8086/8088).

## EL PROCESADOR DE PROPOSITO GENERAL 8088.

El 8088 es un microprocesador de 16 bits de propósito general de Intel . Tiene una unidad de ejecución (EU) y una unidad interfaz de bus (BIU). La unidad de ejecución realiza todas las operaciones mientras que la unidad de interfaz del bus accede a datos e instrucciones del mundo exterior.

El 8088 es internamente un procesador de 16 bits , pero tiene un bus de datos externo reducido de 8 bits.

Una ventaja muy importante del 8088 es que nuevos sistemas, pequeños y baratos aunque muy potentes, pueden diseñarse basándose en este chip.

### 2.1.- CARACTERISTICAS PRINCIPALES DEL 8088

El 8088 tiene un bus de direccionamiento de 20 bits de amplitud , lo que le provee direccionar un megaocteto de memoria esto es  $2^{20}=1.048576=1$  megaocteto. Sin embargo, el registro de direccionamiento del 8088 tiene solamente una amplitud de 16 bits, que son aproximadamente 64k octetos. Esto es posible porque el procesador usa un método llamado segmentación para permitir el direccionamiento a todo el megaocteto de memoria.

El 8088 tiene una memoria separada, llamada espacio de entrada/salida (E/S) con capacidad de 64k octetos, que puede considerarse como una memoria extra para direccionar, en la cual se encuentran los aparatos de E/S (cableadas las direcciones).

## JUEGO DE REGISTROS

El juego de registros del 8088 se compone de 14 registros internos de 16 bits. Cada registro tiene su propia personalidad, aunque algunos comparten tareas comunes.

### MODALIDADES DE DIRECCIONAMIENTO.

El 8088 tiene 25 modalidades de direccionamiento. En cualquier caso son formas de acceder a la posición en la que se encuentra un dato usado durante la ejecución de una instrucción.

### SEÑALES DE RELOJ.

El 8088 requiere una única señal de reloj y depende del chip generador de reloj 8284 que usa un cristal oscilador para determinar la frecuencia de la señal. Cambiando este cristal, se pueden usar diferentes velocidades de operación. Intel tiene una versión de 5 Mhz y otra de 8Mhz para el 8088 que son las velocidades más altas recomendadas para estos chips. No se recomienda bajar de 2 Mhz para ninguna de las versiones. En condiciones ideales el 8088 requiere una señal de reloj que se mantenga a tensión alta una tercera parte del tiempo total del ciclo.

### REQUISITOS DE POTENCIA.

El 8088 requiere solamente una alimentación de 5 volts.

### ENCAPSULADO.

El 8088 se encuentra encapsulado con 40 terminales. En ciertas configuraciones el 8088 puede generar más señales que las terminales que posee. Esto es posible gracias al uso de las técnicas de multiplexado en el tiempo y de codificación.

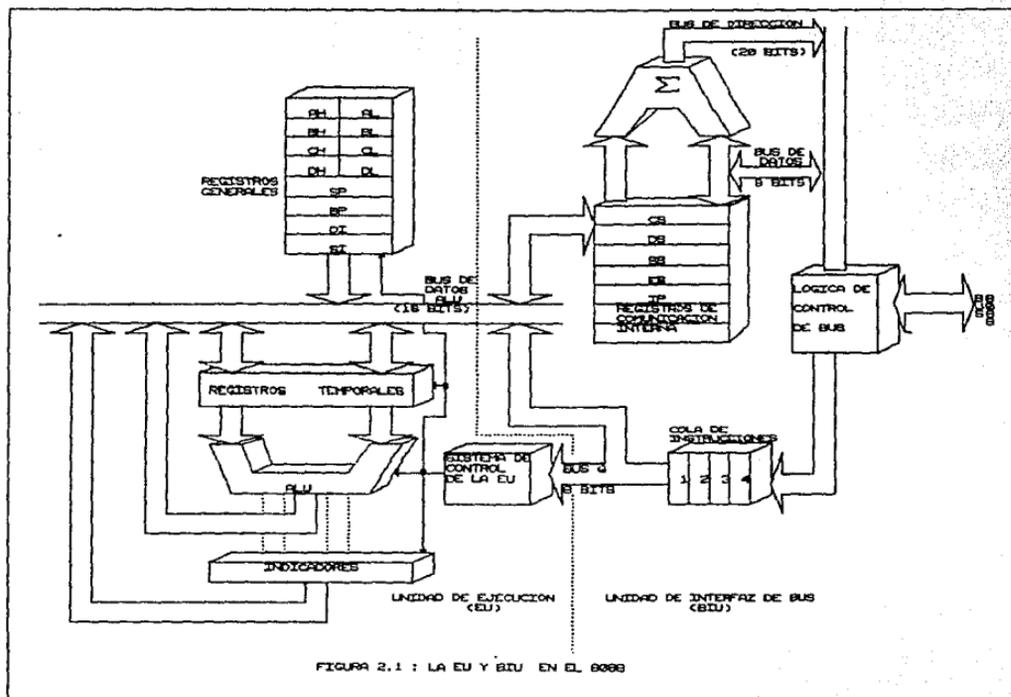
Multiplexado en el tiempo significa usar el mismo conjunto de líneas, pero en períodos de tiempo distintos, para enviar conjuntos de señales diferentes. Codificación significa convertir un conjunto de estados 'posibles en números, y enviar estos números por unas pocas líneas en vez de usar una línea para cada estado posible.

## 2.2.- ARQUITECTURA TUBULAR (PIPELINE).

Intel diseñó el 8088 para realizar al mismo tiempo las principales funciones internas de transferencia de datos y búsqueda de instrucciones. Para lograr esto, el 8088 tiene dos procesadores interconectados en el mismo encapsulado. Una unidad está encargada de buscar instrucciones y la otra de ejecutarlas. Además, la unidad encargada de buscar instrucciones utiliza el método llamado de estructura tubular (pipeline) o por cola para almacenar nuevas instrucciones hasta que se necesiten (Fig.2.1).

Al procesador principal se le llama unidad de ejecución (EU). Está encargado de codificar y ejecutar todas las instrucciones. Se lo llama Unidad de Interfaz de Bus (BIU) al otro procesador. La BIU está encargada de localizar las instrucciones y de transferir todos los datos en los registros y el mundo exterior. La BIU del 8088 debe transferir datos entre el bus de datos interno de 16 bits y el bus de datos externo de 8 bits.

Cuando la BIU localiza en memoria un octeto de código máquina, lo coloca en una línea de espera especial llamada cola



de instrucciones. En el 8088 la cola de instrucciones tiene una longitud de 4 octetos y el código de máquina se guarda de octeto en octeto.

La división del trabajo entre la EU y la BIU permite mayor velocidad al 8088 pues ahorra un tiempo considerable.

### 2.3.- JUEGO DE REGISTROS.

El 8088 contiene 14 registros de 16 bits cada uno. Algunos pertenecen a la EU y otros a la BIU. Los registros de la EU se suelen usar para direccionamiento y son los siguientes :

-Cuatro registros generales de 16 bits (AX,BX,CX,DX), que pueden subdividirse ( y direccionarse separadamente) en ocho registros de 8 bits (AH,AL,BH,BL,CH,CL,DH,DL),en los cuales la X representa "extendido " (16 bits); la H "alto" y la L "bajo". A representa "acumulador"; B,"base"; C,"contador" y D ,"datos" (Figura 2.2).

-Cuatro registros puntero y de índice (SP,BP,SI y DI) los cuales no pueden subdividirse. SP es el puntero de pila, BP es el puntero base ,SI es el registro índice y DI es el registro fuente.

-Un registro de indicadores de 16 bits, que contiene varios bits de estado para el microprocesador. Los indicadores son : de cero (ZF), de signo (SF), de paridad (PF), de acarreo (CR), auxiliar (AF),de dirección (DF), de interrupción (IF), indicador de desbordamiento (OF) e indicador de desvío (TF).

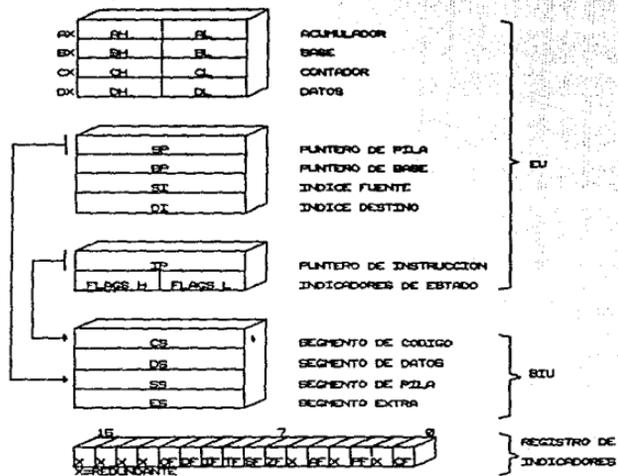


FIGURA 2.2 : JUEGO DE REGISTROS Y REGISTRO DE INDICADORES DEL 8086

-Cuatro registros segmento que son: segmento de código (CS), de datos (DS), de pila (SS) y extra (ES).

-Un puntero de instrucciones (IP).

#### 2.4.- SEÑALES Y TERMINALES

Modo máximo y modo mínimo:

Esta determinado por la conexión del terminal MN/MX a Tierra o Vcc. Con la modalidad máxima el 8088 puede trabajar en conjunción con el Procesador de Datos Numérico NDP8087 y el Procesador de Entrada/Salida IOP 8089, pero se requiere además del Controlador de Bus 8288.

Las señales se pueden clasificar de la siguiente manera:

Alimentación

Reloj

Control y estado

Direcciones

Datos

Los terminales 1 y 20 son para tierra (GND) y la tensión de entrada (Vcc) está en el terminal 40 (figura 2.3).

La señal de reloj se encuentra en la 19

El 8088 tiene un bus de direcciones de 20 bits, de los cuales los 4 más significativos son compartidos con señales de estado, y los 8 menos significativos se comparten con los datos.

S0 a S7 son señales de estado. S0, S1, S2 son accesibles en modo máximo e ingresan al 8288. Algunas de las señales del 8288



las genera el 8088 en modo mínimo.

RD (terminal 32) es una señal que genera el procesador e indica un estado de lectura.

READY (terminal 22) es una señal que entra de los dispositivos externos cuando alguno de estos es seleccionado para lectura o escritura y todavía no está preparado para la transferencia de datos. Pone a cero la línea de señal READY y el procesador añade ciclos de espera hasta que la línea READY regrese al nivel de 5 volts indicando que el dispositivo externo se encuentra listo para la transferencia.

RESET (terminal 21) es una señal de entrada que pasa por el 8284 reinicializa al  $\mu$ P, borra la cola de instrucciones y ciertos registros (los indicadores, puntero de instrucción, segmentos de datos y de pilas y registros de segmentación adicionales). El segmento de código se pone en FFFFH y el  $\mu$ P es forzado a leer FFFF0H en su primer octeto.

En modo mínimo salen las siguientes señales de control y de estado:

SS0 (estado S0, terminal 34)

M/IO (control de memoria/IO, terminal 28)

DT/R (recepción transmisión de datos, terminal 27)

DEN (datos accesibles, terminal 26)

ALE latch de direcciones activo, terminal 25)

WR (control de escritura, terminal 29)

HOLD (terminal 31)

HLDA (terminal 30)

INTA (reconocimiento de interrupciones , terminal 24)

SS0,M/IO y DT/R sirven para determinar el estado en el que se encuentra el 8088: obtención de código, escritura en memoria, escritura ó lectura en puerto de E/S, etc.. DEN y DT/R sirven para especificar al transceptor de bus 8286 donde hay datos y la dirección destino. ALE indica al latch de direcciones 8282 donde encontrar la dirección. WR indica que los datos están disponibles en el bus de datos. HOLD y HLDA son señales de control de bus del 8088 . Cuando es recibida una señal a través de la línea HOLD proveniente de otro aparato que desea acceder al bus del 8088 este último se desconecta eléctricamente del bus y envía la señal HLDA para indicar que el otro aparato puede usar el bus . Al terminar llega una señal por HOLD y el 8088 puede continuar usando el bus.

En el modo máximo salen las siguientes señales :

S0,S1,S2 (señales de estado , terminales 26,27,28)

QS0,QS1 (estado de la cola , terminales 25 y 24)

RQ/GT0,RQ/GT1 (petición/concesión , terminales 31 y 30)

LOCK (terminal 29)

QS0 y QS1 son señales de estado de la cola de instrucciones . El procesador de datos numérico las utiliza para coordinarse con el 8088. RQ/GT0, RQ/GT1 y LOCK operan de la misma forma en que lo hacen la HOLD y HLDA para transferir el control del bus, pero se utilizan para coordinar las actividades

con otros procesadores en una configuración de grupo. LOCK es una señal que comunica a otro procesador del sistema cuando no debe tomar el control del sistema.

La señal TEST (terminal 23) se utiliza para enlazar al 8088 con un procesador paralelo, y sincronizar el proceso principal con los otros.

Las terminales NMI (interrupción no enmascarada terminal 17) e INTR (petición de interrupción terminal 18) son parte del sistema de interrupciones del 8088. El término "no enmascarable" significa que este tipo de interrupción no se puede activar o desactivar vía un comando software. Las interrupciones generadas por INTR pueden desactivarse vía software.

## 2.5.- ESTRUCTURA DE INTERRUPCIONES

Una estructura de interrupción es la que provee el microprocesador para el servicio adecuado de I/O, correcciones y ciertos tipos de errores. Al entrar en una interrupción el  $\mu P$  salva su estado actual, ejecuta una rutina de interrupción y vuelve a reanudar el trabajo. Debe interpretarse la interrupción como una subrutina o llamado de la misma y el programa especial de interrupción es el cuerpo de la subrutina. Las principales diferencias entre subrutinas e interrupciones son:

- 1) Las subrutinas son llamadas únicamente por software, mientras que las interrupciones pueden ser llamadas por hardware o software.

- 2) Las subrutinas guardan la dirección de retorno y las interrupciones guardan además todos los indicadores.
- 3) Las subrutinas necesitan pasar por algún medio, datos del cuerpo principal y hacia este, mientras que las interrupciones no lo necesitan.

La estructura de las interrupciones 8088 utiliza una tabla de 256 posiciones con 4 octetos cada una. Estas posiciones se cargan con un puntero a diferentes rutinas de la memoria principal, 2 octetos son para el segmento de código y 2 del puntero de instrucciones de la rutina. A cada uno de los punteros ( de 4 octetos) se le asigna un número del 0 al 255. Al tipo 0 se le asigna la posición 0 ; al 1 , la posición 4 y así hasta el 1020 siguiendo la fórmula en posición de memoria.

Cada tipo de interrupción puede activarse por hardware ó software y así probar el hardware de interrupción. Si se utiliza la orden de activación de interrupción (SII) se activa la bandera IF de interrupción y al usar la orden de borrar interrupciones (CLE) la bandera IF se desactiva bloqueando las señales de interrupción que entran.

Las interrupciones hardware actúan de la siguiente manera: cuando un dispositivo externo necesita servicio activa una petición de interrupción (INTR) del 8088 . Si el procesador (IF=1) ,envía una señal de recibido en modo mínimo directamente ó por el controlador de bus 8288 en modo máximo.

El dispositivo externo indica mediante un octeto en el bus de datos el tipo de interrupción que desea . El procesador lee el número en la tabla de interrupciones salva sus indicadores y el puntero de pila con la dirección de retorno , y carga el segmento de código y puntero de instrucciones desde la tabla de interrupciones. Es entonces cuando se ejecuta la subrutina de servicio . Al final de ésta debe haber una instrucción de retorno de interrupción IRET para reestablecer los indicadores y entonces el procesador sigue operando normalmente.

Las interrupciones no enmascarables (las cuales no se desactivan mandando a cero IF y son del tipo 2) se reservan para interrupciones de emergencia como falla de potencia o errores de memoria.

Existen interrupciones especiales que se generan al dividir por cero o en capacidades excedidas.

También hay interrupciones especiales para pruebas de programas. Una de estas es la interrupción de paso simple (de tipo 1), la cual actúa interrumpiendo el programa después de que cada instrucción se ejecuta. Otro tipo de interrupción es la de punto de acceso (de tipo 3), la cual tiene un sólo octeto en su código de máquina. Para usarla el programador la inserta en un punto del programa donde desea examinar el contenido de los registros y banderas. Al llegar a este punto el procesador salta a la rutina de servicio 3 otorgando el control al programador para que haga las operaciones que desee.

## 2.6.- MODOS DE DIRECCIONAMIENTO

El 8088 tiene 25 modos de direccionamiento. Son complicados en cierto modo pero pueden tomarse como casos particulares de referencia a registro y referencia a memoria.

En el primer caso el operando está en un registro específico; en el segundo deben sumarse 4 cantidades para encontrar la dirección en memoria: 1) dirección de segmento, 2) dirección base, 3) una cantidad índice y 4) un decalaje.

La dirección de segmento se guarda siempre en el registro de segmentación (DS, ES, SS o CS) y antes de usarse se multiplica por 16 (corrimiento a la derecha de cuatro dígitos binarios). El registro de segmentación siempre se usa para referencia a memoria.

La base se almacena en el registro base (BX o BP). El índice se almacena en el registro índice (SI o DI). Cualquiera de estas cantidades o ninguna puede utilizarse para calcular la dirección real. Estas cantidades son variables ya que al estar almacenadas en registros de propósito general son susceptibles de ser modificadas.

El decalaje es un desplazamiento de 16,8 o 0 bits y es una cantidad estática. Es utilizado para compilar datos, organizar memoria y reubicar más rápida y fácilmente. En el lenguaje ensamblador los parentesis o corchetes indican el modo de direccionamiento.

## 2.7 - ESTRUCTURA DE MEMORIA DE SEGMENTACION

Esta estructura esta pensada de manera que el 8088 pueda acceder a 1 Mbyte con solo 16 bits direccionables.

Las direcciones estan compuestas de dos partes: desplazamiento ( con un decalaje, base e indice) y una direccion segmento (almacenada en uno de los registros de segmentación), el procesador usa estas dos cantidades de 16 bits para calcular la dirección real de 20 bits, según la siguiente fórmula:

$$\text{Direccion real} = 16 * (\text{direccion del segmento}) + (\text{desplazamiento})$$

## C A P I T U L O   I I I

### 3.1.- CIRCUITO BASICO PARA UN SISTEMA CONTROLADOR BASADO EN EL MICROPROCESADOR 8088

Pueden existir configuraciones muy variadas para construir un circuito de control basado en el 8088 o en otro procesador, algunos con mas poder de comunicacion con sistemas exteriores, otros con mayor memoria, mayor velocidad u otras peculiaridades, pero un sistema basico como el deseado es descrito por diagrama a bloques de la figura 3.1, estableciendo sus partes integrantes de la siguiente manera:

- 1.- Circuito de reloj
- 2.- Circuito del CPU ( $\mu P$ )
- 3.- Latches de dirección
- 4.- Latches de datos
- 5.- Memoria ROM
- 6.- Memoria RAM
- 7.- Sistemas I/O

A continuación se explicarán los pasos necesarios para poner en funcionamiento cada etapa, así como el porque de la utilización de cada elemento del circuito.

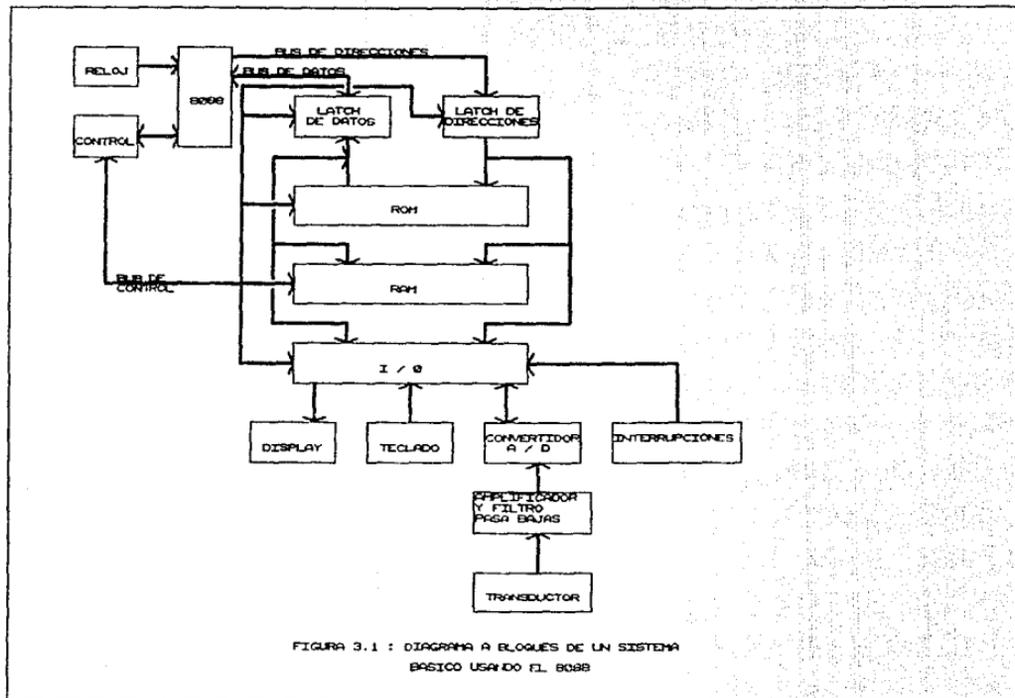


FIGURA 3.1 : DIAGRAMA A BLOQUES DE UN SISTEMA BASICO USANDO EL 8088

### 3.2.- CIRCUITO DE RELOJ.

Para generar los pulsos de reloj requeridos por el 8088 debe usarse un circuito que tenga un ciclo de trabajo en nivel alto del 33% (nominal) con tiempos de subida y de bajada de aproximadamente 10nS (Ver las hojas de datos del 8088 ).

Así es como se propone utilizar un oscilador ya construido ex-profeso para llevar a cabo esta tarea y evitar multiplicar los problemas que se originan al tratar de controlar no un solo C.I. sino varios componentes analógicos o digitales , de un oscilador diseñado a base de elementos discretos.

El circuito 8284 es ideal para tomar bajo su cargo la generación no solo de la señal de reloj sino además de señales de sincronización y de pulsos de reset con las características deseadas.

Para conjuntar el circuito básico que generará las señales necesarias son indispensables los siguientes componentes:

- 1 C.I. 8284
- 1 Cristal resonante serie de cuarzo  
(Frecuencia=14.3181Mhz)
- 1 Diodo 1N914
- 2 Resistores de 510 $\Omega$
- 1 Resistor de 4.7K $\Omega$

- 1 Resistor de 10 K $\Omega$
- 1 Capacitor de 4.7 $\mu$ F
- 1 Capacitor de 12 a 15 pF
- 1 Un interruptor de un polo, un tiro normalmente abierto
- 1 Capacitor de 0.1 $\mu$ F de milar .

El diagrama de conexiones es el ilustrado en la figura 3.2

Las conexiones se explicarán comenzando por el grupo generador de pulsos de reloj.

Las terminales X1 y X2 (terminales 17 y 16 respectivamente) son acopladas a un cristal de cuarzo resonante serie en este caso el 14.3181 obteniendo una salida de reloj (terminal 8) que es un tercio de la frecuencia del cristal (es 4.77Mhz) con un ciclo de trabajo a nivel alto del 33%.

Debido a que la entrada del oscilador del 8284 se comporta como un amplificador es conveniente mantener tan baja como sea posible la resistencia del cristal y por tal razón se le acopla en serie un capacitor de 12pF (hasta 15 pF). Así , se evita que el oscilador comience a saturar la entrada del 8284 ó bien no pueda generar la señal de reloj.



Las resistencias de  $510\Omega$  ayudan a disminuir los corrientes de frecuencia y la capacitancia parásita que se genera especialmente en las tabletas de experimentación, donde los problemas por corrientes parásitas y falsos producen los efectos de inductancia o capacitancia parásita (se sugiere el uso de estas tabletas para el desarrollo de un prototipo experimental antes de pensar en tarjetas de circuito impreso). Estas como el capacitor de  $12\text{ pF}$  pueden ser eliminados del diseño siempre y cuando el montaje del circuito impreso sea cuidadosamente planeado.

Su voltaje de alimentación  $V_{cc}$  en la terminal 18 y nivel de tierra en terminal 9, tomando los requerimientos de suministro TTL ( $5.1\text{ Volts } \pm 10\%$ ).

Pulso para reset manual. Para lograr un pulso aproximadamente  $500\mu\text{s}$  debe usarse un red RC que se cargue a  $V_{cc}$  y se descargue a una constante  $R \cdot C$ , siendo la duración del pulso en la red ilustrada de:

La constante RC del sistema es :

$$T = R \cdot C = 4.7\mu\text{F}(10\text{K}\Omega) = 47\text{mS} \cong 50\text{mS}$$

que es mayor que el tiempo requerido.

La entrada del circuito que usa este pulso es RES (terminal 11) señal de activación negativa. El diodo  $D1$  conectado como se ilustra (en polarización inversa desde  $V_{cc}$  a

tierra) actúa como un dispositivo de seguridad para evitar un corto circuito directo de Vcc a tierra. La resistencia de 150Ω sirve para descargar con seguridad a tierra el voltaje almacenado en el capacitor C4. El capacitor C1 es acoplado para mantener constante el voltaje de alimentación evitando pequeñas caídas o la presencia de rizados en la terminal Vcc (18). Además filtra señales de ruido presentes .

Existen terminales que son puestas a tierra ( terminales 1, 3, 7 y 13). La explicación para esta disposición , que crea un funcionamiento específico , es la siguiente:

CSYNC (terminal 1(Clock sincronization)). Los pulsos que reciba por esta terminal activarán o detendrán el funcionamiento de los contadores internos. CSYNC=1 . Cuando CSYNC=0 podrán verse nuevamente en sus salidas las señales que genera el CI (CLOCK en 8 , PCLOCK en 2 y OSC en 12). Sirve para mantener al 8284 en estado dependiente a una frecuencia maestra que se presenta en la terminal EFI (terminal 14) .

AEN 1 (terminal 3 y AEN 2 , terminal 7. Señales utilizadas como protocolo en sistemas multiusuario. AEN 1 controla la señal RDY 1 (4) y AEN 2 controla a RDY 2 (6) para administrar el tránsito en los buses del sistema al ser ocupados estos por dispositivos distintos al uP.

$\overline{F/C}$ , terminal 13. Se usa para definir como funcionará el circuito, con un cristal de cuarzo y su propio reloj maestro  $\overline{F/C}=0$  ó bien con una señal externa  $\overline{F/C}=1$  que ingrese por EFI .

quedan sin conectar las siguientes señales:

**PCLCK** terminal 2 (peripheral clock). Señal de reloj con una frecuencia de la mitad de **CLCK** (1/6 de la frecuencia del cristal conectado a X1 y X2 ) con la característica de tener un ciclo de trabajo no del 33% sino del 50% en 1 lógico.

**RDY 1**, terminal 4 y **RDY 2**, terminal 6. Utilizadas para control de distintos dispositivos en el acceso a buses. Poniendolos a 1 lógico dejan saber al  $\mu P$  que los buses estan libres para su control.

**Terminales utilizadas en el diseño:**

**READY**, terminal 5. Es utilizada para indicar al  $\mu P$  cuando los demás componentes del sistema estan listos y el  $\mu P$  puede trabajar con normalidad. Con **READY** en 1 se especifica un funcionamiento a plena capacidad . entendiendo que se trabaja con elementos externos lo suficientemente rápidos para la velocidad de trabajo del  $\mu P$  ; por el contrario con **READY** en 0, el  $\mu P$  generará estados de espera (ciclos wait [tw] ) para permitir la sincronización del equipo externo. Esto es posible con la conexión a tierra ó a voltaje de la terminal ASYNC.

ASYNC, terminal 15. Cuando ASYNC esta a 0 se esta trabajando con aparatos más lentos que el  $\mu P$ , con ASYNC en 1 el  $\mu P$  se comunica con aparatos tan rápidos como el mismo .

OSC, terminal 12 .Se obtiene una versión de la frecuencia del cristal a niveles TTL. Es muy util para generar diferentes frecuencias dentro del mismo sistema .

RESET, terminal 10. Alimenta la línea reset del 8088 con un pulso de las características adecuadas para reinicializar al  $\mu P$ .

CLCK, terminal 8. Señal de reloj adecuada a los requerimientos de trabajo del 8088 y otros  $\mu P$  de la familia INTEL; la generación de pulsos obedece a un ciclo de trabajo con 33% en estado alto y 67% en estado bajo. Sus tiempos de caída o subida son de aproximadamente 10nS otorgando una frecuencia que es  $\frac{1}{3}$  de la frecuencia del cristal utilizado .

### 3.3.- INSTALACION DEL MICROPROCESADOR ( $\mu$ P 8088)

En el diseño se proponen la siguientes conexiones:

A tierra ( $V=0V$ ).

**GND 1, terminal 1 y GND 2, terminal 20.** Por construcción el 8088 cuenta con 2 terminales de tierra las cuales NO ESTAN CONECTADAS INTERNAMENTE como se utiliza en algunos circuitos. Por eso deben ponerse ambas a tierra si se quiere evitar dañar al  $\mu$ P.

**NMI, terminal 17.** En este nivel del diseño no se prevee utilizar interrupciones no mascarables cuya forma de activar es a 1 lógico. No se le puede desactivar hasta que se ejecuten las rutinas que se piensen utilizar con una interrupción no mascarable ó se efectue un regreso de interrupción (IRET). Su estado normal es 0.

**INTR, terminal 18.** Esta terminal al ser activa en 1 lógico avisa al  $\mu$ P de la presencia de una interrupción y la subsecuente ejecución de software creado para la misma. Normalmente a 0 es la vía de utilización del servicio de interrupciones .

**TEST, terminal 23.** Normalmente a 0 debido a que obedece la ejecución de una instrucción WAIT (que la lleva a estado alto), hasta que el hardware conectado a ella lo lleva a 0 reiniciando el  $\mu$ P su funcionamiento .

**HOLD, terminal 31:** Usada para dar por enterado al  $\mu P$  de la ocupación del bus del sistema, para acceso a memoria DMA, por ejemplo u otras actividades que liberan al  $\mu P$  de trabajo que disminuya su velocidad de ejecución. Esta señal es utilizada en sistemas de multiprocesadores de acceso a un mismo sistema de buses y la misma memoria.

A voltaje Vcc:

**VCC, terminal 40:** Alimentación a 5.0V  $\pm 10\%$  (lleva un condensador que filtra la señal para mayor pureza del voltaje alimentado).

**MIN/MX, terminal 33:** Se establece que el  $\mu P$  8088 trabaje en modalidad mínima. Esta terminal determina el tamaño del sistema en el que opere el  $\mu P$ . Con MIN/MX en 0 lógico se utilizan todas las señales del 8088 generadas utilizando el decodificador de señales de status 8288 obteniendo las siguientes señales de control:

- 1 Instrucción fetch (adquisición de instrucción)
- 2 Read memory (lectura de memoria)
- 3 Write memory (escritura de memoria)
- 4 No activity (sin actividad)
- 5 Interrupt acknowledge (reconocimiento de interrupción)
- 6 Read I/O port (lectura de puerto)

- 7 Write I/O port (escritura de puerto)
- 8 Halt (detencion)

La modalidad máxima es utilizada para trabajo conjunto con el coprocesador NDP 8087 y otros CI de soporte de la familia Intel.

Con MIN/MX en 1 se utiliza el 8088 en su modo mínimo para trabajo en sistemas que tienden a control muy específico y de tamaño mínimo.

**READY**, terminal 22 .Alimentada del 8284 PIN 5

**CLCK**, terminal 19 .Alimentada del 8284 PIN 8

**RESET**, terminal 21 .Alimentada del 8284 PIN 10

**CLCK** es alimentada a 4.77Mhz , siendo esta una frecuencia adecuada para el 8088-2 (rango de 2 a 8 Mhz) tomando en cuenta la relación costo-velocidad de funcionamiento.

El diagrama de conexiones se muestra en la figura 3.3.

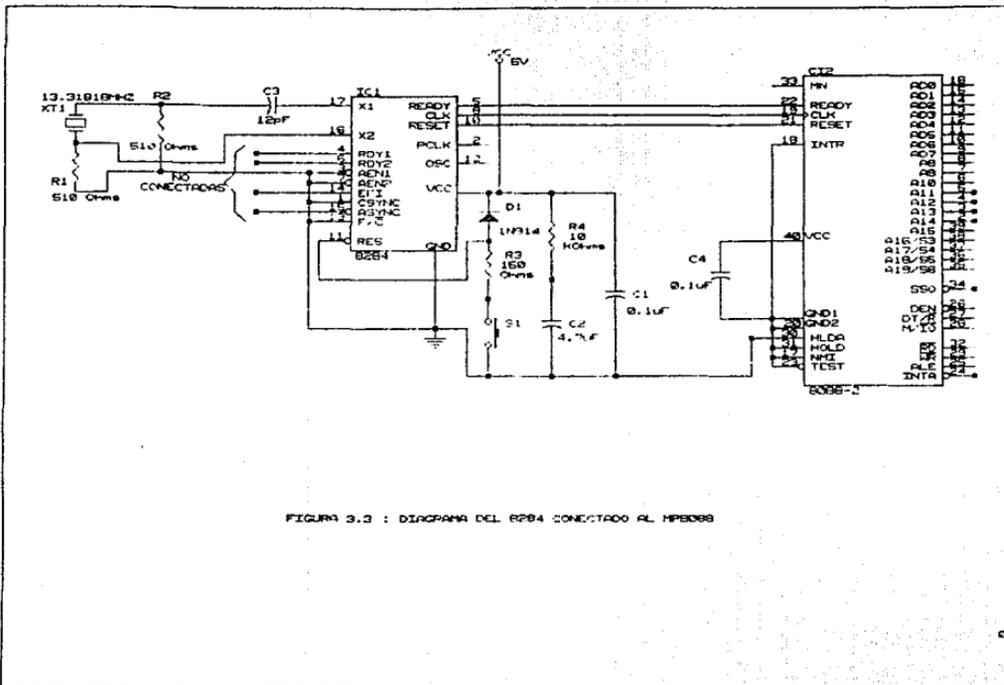


FIGURA 3.3 : DIAGRAMA DEL 8284 CONECTADO AL HPB088

### 3.4.- DEMULTIPLEXION DE BUSES

Para demultiplexar el bus de datos y de direcciones se sugiere un arreglo sencillo de conectar, donde se pueden explorar las señales en caso de error de manera rápida. A esto se agrega la ventaja de proteger el  $\mu P$  de daños por posibles cortocircuitos o descargas debidas a accidentes o pruebas en los buses de direcciones o de datos, agregando que no existen problemas de sincronización crítica.

Las conexiones son las señaladas en la figura 3.4.

Los elementos que deben instalarse junto a los ya existentes son:

- 2 C.I. 74LS373                      Latches octales tipo D
- 1 C.I. 741s245                      Tranceptores de bus octal
- 3 Capacitores de 0.1 $\mu$ F (Cualquier voltaje superior a 5)

Los 3 capacitores conectados a las terminales de voltaje filtran la señal de Vcc (on una conexión práctica debe explorarse los valores de capacitancia y voltaje que mejor purifiquen la señal de C.D.), obteniéndose mejores niveles lógicos sin pérdida de la información).

Se utiliza equipo LS debido a su velocidad de respuesta y sus mejores propiedades de estabilidad térmica y contra el ruido, en comparación de otras versiones TTL.

#### Conexiones al 8088

$DT/\bar{R}$ , terminal 27 a la terminal 1 del 74LS245.  
Habitualmente usada para poner "sobre aviso" a aparatos más lentos que el  $\mu P$ . Activa en una dirección o a la inversa a una dirección del latch :

$DT/\bar{R} = 1$  Transmisión de datos a partir del 8088.

$DT/\bar{R} = 0$  Recepción de datos.

ALE, terminal 25 (Habilitador del latch de direcciones). Activa con anticipación a una dirección válida los latches estableciendo la salida de tal dirección al poner a 1 lógico la señal en la terminal 11 de los 74LS373 .

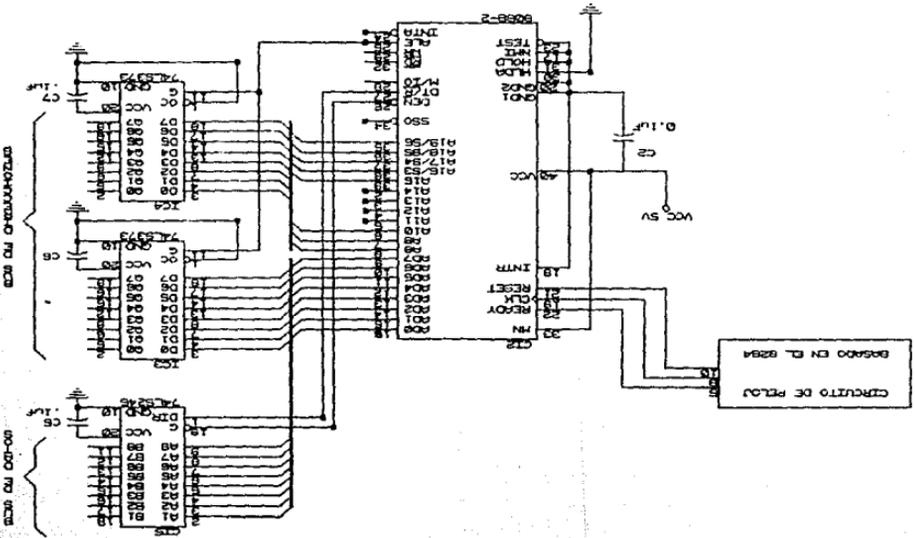
AD0 - AD7, terminal 16 a terminal 9. Utilizadas en tiempo compartido como datos o direcciones. Veanse conexiones en la figura 3.4.

A8 - A10 y A15 - A19, terminal 8 a 6 y terminal 39 a 35.  
Son utilizadas para definir primero el mapa de memoria del 8088 utilizando entonces a A10 como conmutador entre dos tipos de memoria de 11 bits del bus de direcciones (AD0 a A10) capacidades de 2K bytes de ROM con A19 = 1 y 2K bytes de RAM con A19 = 0 .

Señales sin conectar:

A14 - A11 no han sido utilizadas (en esta etapa del diseño).

FIGURA 3-4 : DEMULTIPLEXION DE DIRECCIONES Y DATOS



**INTA**, terminal 24. Reconocimiento de interrupción , no es usada en este momento. Se utiliza para reconocer la activación de **INTR** por un aparato externo .

**HOLDA**, terminal 30. Reconocimiento de ocupación de bus por un equipo externo , útil para DMA , refresco de RAM dinámica y otras tareas.

**SS0**, terminal 34. Status 0 , señal de reconocimiento de estado . Solo útil cuando el  $\mu P$  esta conectado en modo MAX .

$\overline{WR}$  terminal 29 ,  $\overline{RD}$  terminal 32 y  $IO/\overline{M}$  terminal 28. Por el momento no son conectadas .

$\overline{DEN}$ , terminal 26. Conectada a la terminal 19 del 74LS245 (OE) . Es la que permita la ejecución de una transferencia de datos en la dirección adecuada .

En este punto es posible vigilar señales como **CLCK** y **RESET** otras como **ALE** ó  $\overline{DEN}$  no es posible visualizarse porque son generadas durante los primeros ciclos de transmisión de datos o de direcciones (ALE se activa en el ciclo T1 de cada ciclo de dirección ;  $\overline{DEN}$  lo hace cuando un dato esta presente para lectura ( $\overline{RD}$ ) ó de escritura ( $\overline{WR}$ ) .

### 3.5.- INSTALACION DE MEMORIA RAM Y ROM

Un sistema de microproceso como el diseñado necesita donde guardar variables y programas. Para el nuestro se ha pensado con un area básica de RAM y otra de ROM. Prácticamente se puede disminuir el volumen de la memoria se desea usar una 6116 con 2K bytes de memoria , hay C.I. con menor memoria (en el mercado mexicano) ; pero en ROM no es posible debido a que el volumen mínimo en el mercado es de 2K bytes de una EPROM 2716 cuya ventaja es que puede programarse un cierto número de veces previendo así la ejecución de software de diagnostico antes de crear el software permanente del controlador (firmware) .

Con el tiempo es posible aumentar el tamaño de memoria ROM a una 2732 (4K bytes) ó una 2764 (8K bytes) segun el volumen de los programas que se instalen .

Para realizar la distribución de memoria RAM y ROM debe realizarse una selección de señales que establezcan que tipo de memoria esta llamando el 8088 .

El 8088 es un  $\mu P$  que se caracteriza por buscar su primera instrucción en la parte alta de su mapa de memoria ; el Z80 hace en la localidad absoluta 0000h pero el 8088 lo hace en su "localidad de encendido"FFFF0h quedando en el fin de mapa de memoria solo 15 bytes .Algunos fabricantes utilizan estos 15 bytes para dar la fecha de diseño del BIOS de sus computadoras .

Los programadores que usen este tipo de  $\mu P$  de ejecución "de arriba hacia abajo" utilizan saltos a localidades de memoria más bajas donde ya pueda utilizarse un volumen de almacenamiento adecuado. Esta es la forma en que se manejará el mapa del controlador, con un salto a el principio de la EPROM. Estos saltos serán direccionados por etiquetas evitando de manera sencilla utilizar directamente los elaborados esquemas de direccionamiento segmentado del 8088, muy útiles para paginación lógica, acceso a memoria en sistemas basados en microprocesadores de la familia Intel 8086.

Un ejemplo de direccionamiento absoluto se da a continuación:

Se tiene uno de los registros de segmento en C800h  
 Se ejecuta una instrucción o se obtiene un dato de 00BDh  
 (que es un offset dentro del segmento)

El  $\mu P$  hace lo siguiente :

Dirección del segmento	C800h
Lo corre a la izquierda 4bits :	C8000h
O bien lo multiplica por 16d ó 10h.	
Suma el valor del offset :	<u>00BDh</u>
	C80BDh

Se obtiene una dirección real dentro de un mapa de memoria de un Megabyte, direccionable por 20 bits, a partir de registros de 16 bits.

Un ejemplo más complejo :

BX = 4000h Registro base .

DI = 20h Índice de destino .

DS = 3000h Segmento de datos situado en 3000h .

offset= 6h Offset dentro del segmento .

DISP (desplazamiento) = 6h + BX + DI

DISP = 6h + 4000h + 20h = 4026h

Dirección real = 10 \* DS + DISP

Dirección real = 3000 \* 10 + 4026h = 34026h

Esto el  $\mu P$  lo hace automáticamente dentro de su unidad BIU. Esto puede ser aprovechado de la siguiente manera :

La memoria RAM se emplaza a partir de la dirección absoluta 00000h para 2Kbytes de RAM en la parte baja del mapa. Solo A0 a A10 son significativos para ella ; de A11 a A19 no las detectas y realmente al direccionar datos en la RAM (6116) de A11 a A19 permaneceran a 0 lógico . Para ROM es más elaborado. Como el  $\mu P$  comienza a funcionar a partir de los últimos 15 bytes direccionando por etiquetas desde FFFF0h a una localidad más baja , direccionando 2K bytes de memoria ROM desde FFFF0h hacia "abajo" se obtiene la dirección FF800h (dirección 0000h de la EPROM) : a partir de allí se ejecutará el software (utilidades y software de diagnóstico). Así , se establece que para todo el

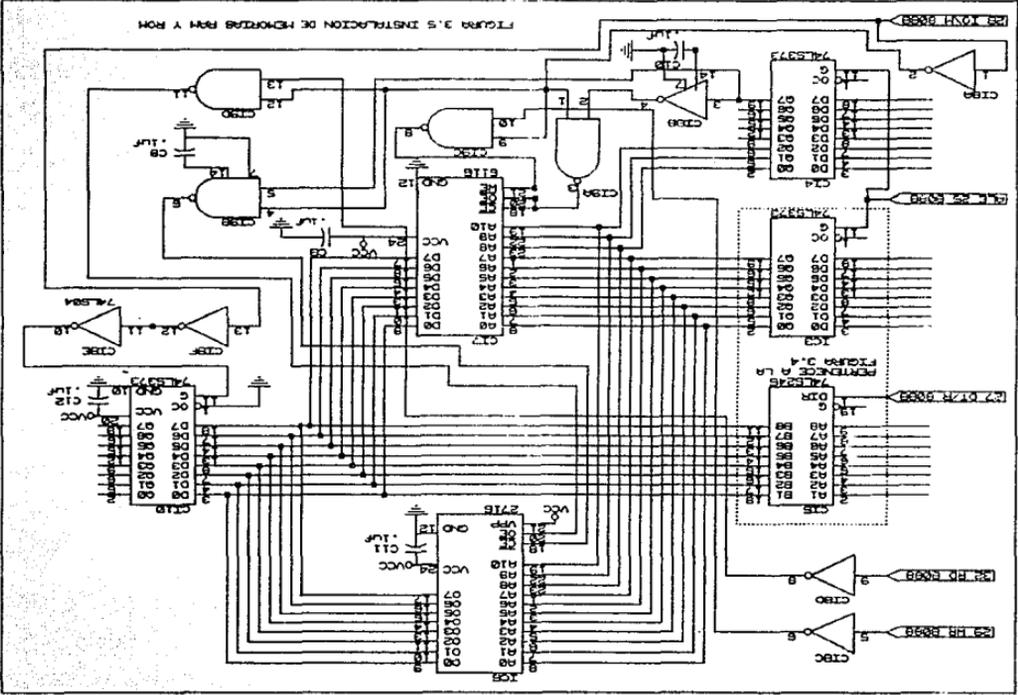
software de EPROM el  $\mu P$  direcciona todas las instrucciones en la parte alta del mapa y a pesar de que la 2716 solo utiliza de A0 a A10 , de A11 a A19 también se activarán , usándose las más significativas (para mayor seguridad y no cambiar el emplazamiento del hardware a corto plazo ) de las direcciones A19 , para conmutar entre RAM y ROM .

Las conexiones se muestran en la figura 3.5 y en la tabla 3.0 se encuentran las señales para la lógica de control de las memorias.

TABLA DE VERDAD DE LOGICA PARA CONTROL DE MEMORIAS									
ENTRADAS				SALIDAS					
IO/M	A19	WR	RD	RAMSEL	RAMRW	ROMSEL	ROMRD	IOSEL	COMENTARIOS
0	0	1	1	0	1	1	0	0	LEER RAM
0	0	0	1	0	0	1	1	0	ESCRIBIR RAM
0	1	1	0	1	1	0	0	0	LEER ROM
0	1	0	1	1	0	0	1	0	NO OPERA
1	x	x	x	1	1	1	1	1	I/O OPERA

TABLA 3.0

FIGURA 3.5 INSTALACION DE MEMORIA RAM Y ROM



### 3.6.- PUERTO DE ENTRADA/SALIDA

El 8088 tiene un inmenso espacio de entrada/salida. Puede direccionar 65536 diferentes puertos. La estructura de interrupciones del 8088 utiliza una tabla de 256 posiciones de 4 octetos cada una, la cual está al inicio absoluto de la memoria (00 a FFh) justo como en las memorias una sola terminal en el 8088 permite conocer si se lee o escribe en una localidad de memoria o en un puerto de entrada/salida. Para acceder a un puerto la línea IO/M debe estar en alto, ser puesta la dirección del puerto en el bus y finalmente activar alguna de las líneas RD o WR. Es bueno tener un lote de puertos de entrada/salida en el sistema, pues da flexibilidad, poder y la oportunidad de controlar cualquier cosa que se desee. Es necesaria lógica de control para la I/O igual que en la memoria.

El bus de dirección es el único camino que tiene el 8088 para decir con que puerto se desea comunicar. Cuando el 8088 se comunica con un puerto solo ese puerto debe estar conectado al bus. Se necesita lógica para decodificar señales del 8088 y proveer dos líneas, una para indicar escritura en I/O (IOWR) y otra para indicar lectura de I/O (IORD). El procedimiento para hacer esto es similar al usado para generar con compuertas las señales RAMRD, RAMWR, ROMSEL y ROMRD. La señal IORD es necesaria para cuando se desee leer en un puerto. En la tabla No. 3.1 se muestra la tabla de verdad para crear las señales necesarias y el diseño con compuertas en la figura 3.6.

Tabla de verdad para la logica de control de I/O					
ENTRADAS			SALIDAS		
IOSEL	RD	WR	IORD	IOWR	Comentarios
0	1	0	1	0	Escritura en I/O
0	0	1	0	1	Lectura de I/O
0	0	0	-	-	Estado imposible
0	1	1	-	-	Estado imposible
1	X	X	1	1	Operación de memoria

Tabla 3.1

Se pueden hacer los puertos utilizando el Controlador Programable Paralelo de Interfaz 8255 (PPIC ó PPI). La transmisión paralelo es útil en todas aquellas aplicaciones que requieran una gran velocidad. Los datos se envían tan rápidamente, o tan lentamente como lo permita el software. Si la transmisión debe ser lenta se introducen retardos por software. El 8255 viene en un encapsulado de 40 terminales, tiene 24 líneas de entrada/salida (3 puertos) para las cuales el usuario puede definir gran cantidad de configuraciones y un puerto de control. Escribiendo diferentes valores en el puerto de control se pueden definir las funciones de los otros tres puertos. Hay tres modos de transmisión básicos : el modo 0 para entrada/salida básica, el modo 1 para entrada/salida habilitada, y el modo 2 para bus direccional. Ya que las características de los puertos dependen de la palabra de control cargada en el registro comando el 8255 puede ser configurado dinámicamente vía software.



Como el 8088 puede direccionar más de 65000 puertos , una parte básica del subsistema de entrada/salida es la lógica para hacer seguro que el 8088 esta conectado al puerto correcto cuando hay operación de I/O. Se debe definir la localización de los puertos como primer paso , se supone que correspondan los cuatro primeros puertos. con los cuatro primeros puertos en el mapa, esto es , los puertos No. 0 al No. 3. Los tres primeros puertos del 8255 son programables de propósito general y el ultimo el numero 3 es el registro comando del 8255.

Para que el 8088 pueda ser capaz de comunicarse en orden con alguno de los puertos .Es necesario adicionar al circuito señales que tienen que ser generadas y enviadas a las terminales de control del 8255, es necesario controlar la terminal "chip select" del 8255. Un 1 lógico en la línea selector de chip causa que las terminales de datos esten flotando, y esto es necesario para evitar contención en el bus de datos. El circuito debe poner en bajo la terminal selectora de chip del 8255 siempre que el 8088 desee comunicarse con alguno de los cuatro primeros puertos. El circuito que realiza esta tarea es un selector de datos que vigile las tres líneas más bajas del bus de direcciones (A0 a A2) y si otras condiciones son satisfechas que ponga en bajo el selector de chip del 8255. Cuando el 8088 da una orden para direccionar uno de los puertos del 8255 sucede lo siguiente:

- 1.La dirección del puerto es puesta en el bus de direcciones.

2. La línea  $\overline{IO/M}$  es puesta en alto para indicar operación de entrada/salida.

3. alguna de las líneas  $\overline{RD}$  o  $\overline{WR}$  del 8088 es activada.

Si el 8088 escribe en un puerto (o si otra cosa se desea que suceda también) el dato deseado es puesto en el bus de datos del sistema.

Uno de los circuitos que se pueden utilizar es el 74LS259 que es un latch direccionable. Conectando la terminal reset del 259 con la línea A7 del 8088 y su terminal habilitadora con la línea  $\overline{IOSEL}$ . El 259 puede trabajar como un decodificador de 3 a 8 y seleccionar una salida por decodificación de las tres líneas más bajas del bus de direcciones (A0 a A2). El 259 es conectado para poner en bajo todas sus salidas no seleccionadas y debe conectarse la terminal D hacia V+. Con lo anterior se consigue que la salida seleccionada sea puesta en alto y se distinga de las no seleccionadas. Ahora se tiene al 259 decodificando las direcciones de entrada/salida pero se necesita un bit lógico adicional para controlar al 8255. Con el 259 se seleccionan hasta ocho salidas, siempre que alguna de las primeras cuatro salidas del 259 vaya a alto se debe habilitar el 8255 (hay que recordar que se tiene mapeado el 8255 como los primeros cuatro puertos del sistema), con un arreglo con compuertas NOR y NAND se puede lograr esto.

### 3.7.- DISPLAY

Los sistemas de display diseñados con dígitos de siete segmentos tienen que tomar información binaria como una entrada y convertir esta a señal de salida que encienda el segmento correcto en el display. Esto se puede realizar con un decodificador. Los decodificadores de LED estándar tienen que acoplar los datos de una tabla interna y cambiar el dato binario de entrada a el caracter modelo de salida, los decodificadores trabajan entonces en forma similar a las memorias. Si se tiene un diseño de mapas de memoria y de entrada/salida también se tiene un diseño de mapas de display. Para trabajar con los datos binarios se requieren como mínimo cuatro líneas de entrada, pero si se utilizan cinco se pueden tener hasta 32 diferentes modelos de caracteres en la salida.

Se puede conectar el display en uno de los puertos de entrada/salida, tres líneas pueden usarse como control para multiplexar el display y pueden controlar ocho dígitos, las cinco líneas restantes se pueden conectar al generador de caracteres.

Utilizando una memoria eprom 2716 para el generador de caracteres, sus líneas de direcciones deben servir como entradas y estar conectadas al bus de datos del sistema, las terminales de datos de la 2716 deben estimular los segmentos del display. Lo primero que se tiene que hacer son las asignaciones de las terminales (decidir qué terminal va a excitar cada segmento) y lo segundo es decidir que caracter se desea generar. Toda esta

información se muestra en la tabla No. 3.2. Usando display de catodo común, se necesita un 1 para encender los segmentos deseados y un 0 para los demás.

Tabla de verdad del generador de caracteres											
Entradas						Salidas					
Dato binario	Dato Hex.	D7 DP	D6 G	D5 F	D4 E	D3 D	D2 C	D1 B	D0 A	Dato Hex.	Leds encendido
00000	00	0	0	1	1	1	1	1	1	3f	111111
00001	01	0	0	0	0	0	1	1	0	06	11110
00011	03	0	1	0	0	1	1	1	1	4F	11101
00100	04	0	1	1	0	0	1	1	0	66	11011
00101	05	0	1	1	0	1	1	0	1	6D	11010
00110	06	0	1	1	1	1	1	0	1	7D	11001
00111	07	0	0	0	0	0	1	1	1	07	11000
01000	08	0	1	1	1	1	1	1	1	7F	10111
01001	09	0	1	1	0	1	1	1	1	6F	10110
01010	0A	0	1	1	1	0	1	1	1	77	10101
01011	0B	0	1	1	1	1	1	0	0	7C	10100
01100	0C	0	0	1	1	1	0	0	1	39	10011
01101	0D	0	1	0	1	1	1	1	0	5E	10010
01110	0E	0	1	1	1	1	0	0	1	79	10001
01111	0F	0	1	1	1	0	0	0	1	71	10000
10000	10	0	0	0	0	1	0	0	0	08	11111
10001	11	1	0	0	0	0	0	0	0	80	11110
10010	12	0	1	0	0	0	0	0	0	40	11101
11111	FF	0	0	0	0	0	0	0	0	00	11100

Tabla 3.2

El número a ser desplegado se envía en los cinco bits más bajos y el display en el que el dígito se enciende es seleccionado con los tres bits restantes. El display puede ser multiplexado con el circuito 74LS138 (que es un decodificador de tres a ocho con tres entradas habilitadoras) para habilitar el display todo el tiempo con encendido individual de cada uno de los dígitos. Las terminales Y0 a Y7 son las líneas habilitadoras de cada uno de los ocho dígitos (Fig 3.7). En la figura 3.8 se muestra el diagrama de los latch para los displays.

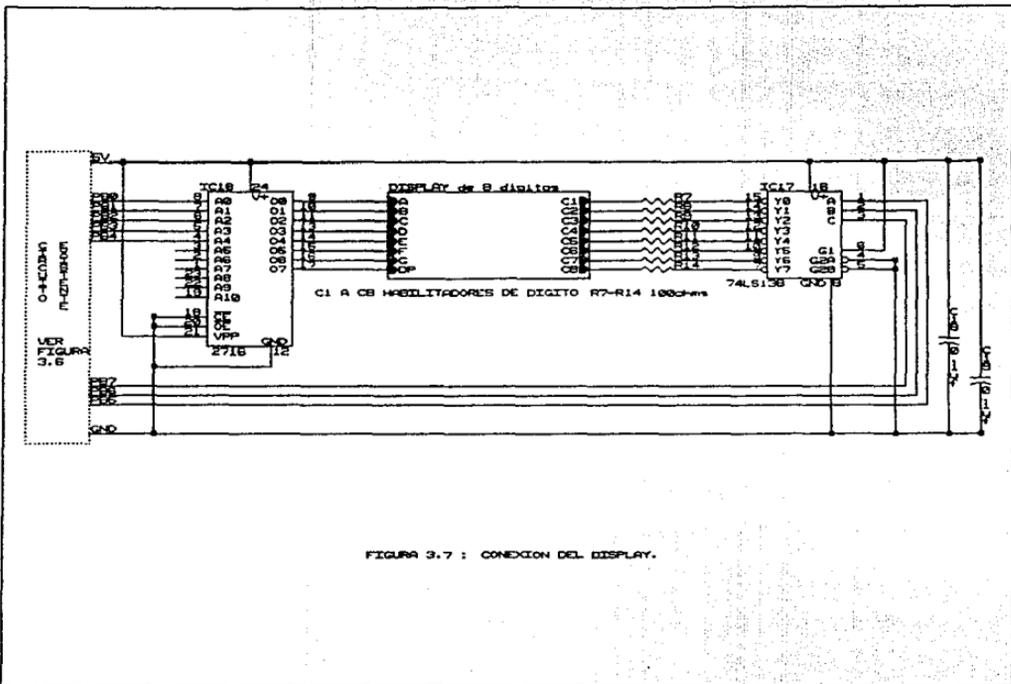


FIGURA 3.7 : CONEXION DEL DISPLAY.

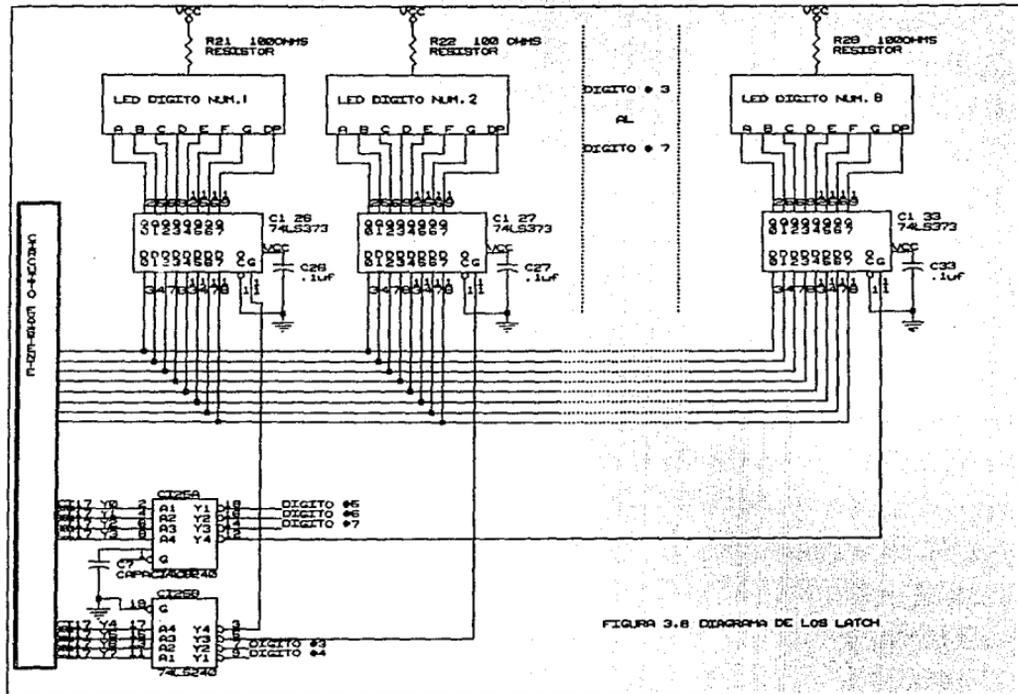


FIGURA 3.8 DIAGRAMA DE LOS LATCH

### 3.8.- TECLADO PARA EL SISTEMA CONTROLADOR

Se ha hablado de la velocidad y poder que un controlador basado en el  $\mu P$  8088 puede mostrar, aun su capacidad de reconfigurarlo por software programado en una EPROM, lo hace ideal para casi cualquier tarea. Pero existe una característica de la que depende para realizar sus funciones: la adquisición de datos de puerto. Enviar datos a puerto es la principal manifestación de trabajo en la que se podría pensar que realiza el sistema y que adquirirlos se limita a apuntar a una dirección en memoria ROM direccionada por el registro IP, ejecutar una instrucción y mandar datos a una pantalla o a la memoria RAM. Pero para aprovechar aún más la capacidad de trabajo de un sistema con  $\mu P$  es necesario contar con la posibilidad de modificar el software "sobre la marcha", es decir, que se puedan reajustar parámetros, marcar rutas distintas de ejecución de labores, entrada de condiciones nuevas de funcionamiento y muchas otras operaciones. Así es natural pensar en la forma que un usuario potencial se comunique con este sistema por medio de un teclado.

Un teclado es un de tantos periféricos con los que se puede explotar la gran capacidad de reconfiguración operativa que tiene un sistema.

Dentro del diseño de la balanza electrónica controlada por el  $\mu P$  8088 las ventajas de conectar un teclado son las siguientes:

ACCESO O SELECCION DE MENUS. Para un firmware más complejo, con un teclado puede establecerse por menus (como de software de una computadora ) la configuración completa de la forma de trabajo de la balanza, nivel de, precisión de datos ofrecidos, forma de presentarlos ó hasta determinar si se desean almacenados en la RAM del sistema ó mandarlo a dispositivos externos (posiblemente conectando una interface paralelo o una RS232C a una computadora).

Establecemos entonces la necesidad de conexión de un teclado. Pero debe desarrollarse tanto el hardware como el software, en caso de una construcción real de este equipo, para llevar a cabo tres procesos con los que un teclado se convierte en interfaz de un sistema de control.

1.- Como codificar cada tecla .

Debe ofrecerse o bien un código binario o cualquier otra manifestación eléctrica particular de cada tecla pulsada.

2.- Como comunicarse con el sistema.

Un teclado es esencialmente un juego de interruptores mecánicos que deben conectarse a un controlador que maneje señales lógicas basadas en voltajes que aparecen tanto a la salida como a la entrada de las terminales del teclado las cuales son características y unicas para cada tecla pulsada .

### 3. Como manejar los datos del teclado .

El bien obtener datos del teclado es primordial. dentro del controlador hay que establecer que valor se les debe dar ó el significado de cada código

Para poder acceder con bases firmes al desarrollo de comunicación entre el teclado y el sistema además de saber que hacer con los datos, primero hay que determinar como debe ser el teclado.

Hay dos formas de construir un teclado. Como un juego de switches que se alimentan y activan líneas formando , ya sea una matriz XY o el mismo juego de interruptores con una línea común que sirve para mostrar a un puerto el código teclado.

La elección de que tipo de teclado se use representa la evaluación de los componentes y señales ya existentes en el sistema y la facilidad de encontrar otros elementos accesorios. Otro factor importante es el número de teclas que se deseen codificar.

El tipo de teclado puede elegirse rápidamente, ya que a mayor número de teclas un teclado de matriz XY, se convierte en la elección adecuada, debido al menor número de líneas que se deben codificar en comparación con el teclado de línea común, que para el mismo trabajo requiere no sólo de una línea por tecla sino además de un mayor número de componentes discretos.

El número de líneas para un teclado de línea común crece casi de manera geométrica respecto al número de líneas y componentes requerido en el teclado de matriz.

El teclado como ya se dijo es una colección de interruptores normalmente abiertos, instalados en un tablero según el gusto preferencia o las posibilidades económicas; el teclado puede comprarse, contruirse o mandarse construir especialmente para determinada presentación. Pueden obtenerse teclas sueltas, teclados comerciales en presentaciones estándar más comunes o contruidos en los materiales y configuraciones totalmente adecuadas a las necesidades del cliente.

Para fines totalmente prácticos se desea utilizar un teclado de matriz de al menos 16 teclas (en matriz son 4 x 4 líneas) para realizar las primeras pruebas de escritura por puerto y tranferencioas a pantalla. Otra razón para el número de teclas es que el código hexadecimal tiene 16 símbolos 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, F con los que se puede escribir directamente en la memoria RAM del sistema, programas en lenguaje ensamblador si es creada una rutina para ello.

En la actualidad existen teclados de matriz contruidos en distintas formas, pero son esencialmente dos: teclados de botones mecánicos (teclas con resortes o muelles) y teclados de membrana. Un teclado usado para fase experimental de hardware

debe ser muy resistente para que no sea necesario renovarlo continuamente. No pueden trabajarse problemas de diseño o programación cuando una de las partes periféricas es susceptible de dañarse retrasando el desarrollo de un proyecto. La experiencia marca que los teclados de membrana , a pesar de ser estéticamente agradables , de una sola pieza y muy compactos , no pueden utilizarse de manera continua antes de que las membranas plásticas ó de hule que separan dos contactos hechos con una película de carbón , se rompan o pierdan su consistencia ocasionando falsos contactos , que se peguen las teclas ó ruptura absoluta del conjunto que forma el botón .

Los teclados mecánicos raramente se descomponen , es necesaria mucha rudeza para provocarlo y pueden ser comprados como un conjunto ó en piezas sueltas para dar el tamaño deseado a esta interface .

La elección recomendada, para esta etapa del proyecto es un teclado de matriz de 16 teclas (4X4) mecánicas de resortes.

Un teclado de 4X4 líneas (16 caracteres) tiene 4 líneas que pueden llamarse "columnas" y 4 que pueden denominarse " renglones". Cualquiera de los dos conjuntos pueden activarse como entrada ó salida , pero ambos se muestrean en su bus al exterior para crear un código de 8 bits (8 líneas) único para cada tecla .

### COMO VISUALIZAR UN CODIGO UNICO PARA CADA TECLA.

Es sencillo hacer que las salidas muestren una a la señal de voltaje pero esto implica , primero , tener permanentemente a Vcc las entradas , desperdiciando 4 líneas ó bits y obteniendo solo 4 códigos .Vease la tabla 3.3 .

D7	D6	D5	D4	D3	D2	D1	D0
1	1	1	1	0	0	0	1
1	1	1	1	0	0	1	0
1	1	1	1	0	1	0	0
1	1	1	1	1	0	0	0

Tabla 3.3

Hay que mostrar una señal de voltaje en solo una entrada para obtener unicamente una salida al momento de muestreo de las 8 líneas , para que cada tecla tenga un código único .

Si cada tecla de la entrada registra un pulso secuencial de voltaje , sus cuatro posibles salidas de cada entrada estaran durante el tiempo que dure el pulso , en posibilidad de presentar voltaje. Después del tiempo del pulso las mismas 4 salidas podrán ofrecer señal de voltaje si se cierra el circuito , pero para otras cuatro combinaciones .

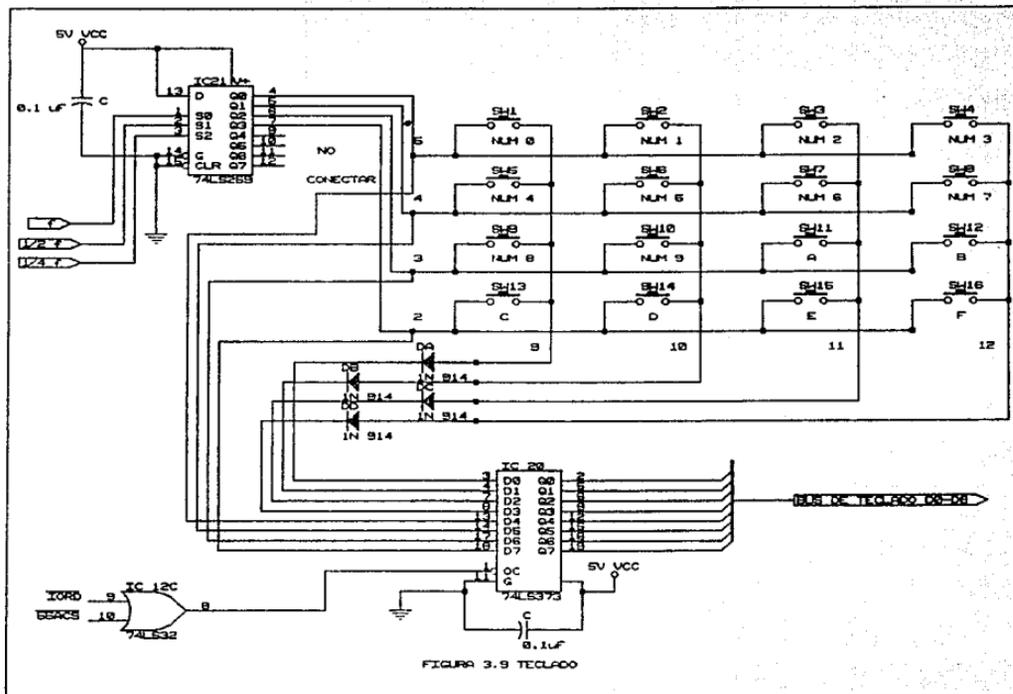
Una secuencia de pulsos de voltaje se puede lograr con registros de corrimiento , flip-flops o lógica combinacional alimentada por señales de reloj sincronas de diferentes frecuencias.

Si se alimenta con  $V_{cc}$  la entrada de un multiplexor y sus selectores con diferentes frecuencias acopladas (la frecuencia al bms sería  $f_1$ , el bit con el siguiente peso sería  $1/2f_1$  y el BMS tendrá una frecuencia de  $1/4f_1$ , en un multiplexor de  $3 \times 8$  se tendrá a la salida 8 pulsos, que pasaran de  $D_0$  a  $D_7$ , de los cuales pueden utilizarse sólo de  $D_0$  a  $D_3$ . Las entradas para  $D_0, D_1, D_2$  y  $D_3$  muestran un pulso que las recorre en secuencia, pudiendo muestrear 8 bits de datos.

En cualquier interface mecánica, una de sus desventajas es introducir una gran cantidad de ruido, picos de voltaje, o incluso cortos a tierra, esto puede, por lo menos, quemar las entradas del PPI, entonces es necesario protegerlo con un separador de señales como un buffer octal 74LS373 quedando el circuito ilustrado en la figura 3.9.

Los diodos conectados, en las salidas sirven para evitar un regreso de voltaje de salida a salida y al ser conectados a una misma línea sirven además de dato, como una señal de estrobo del habilitador del buffer (74LS373).

La terminal 1 del buffer octal está conectada a una compuerta OR que la activará cuando estén presentes cualquiera de las 2 señales  $\overline{IORD}$  (lectura de puerto) ó  $\overline{SSACS}$  (habilitador del CI 8255) permitiendo la transferencia de datos del teclado al PPI vía el 74LS373.



El problema siguiente es la interpretación de los datos. Cada código para un teclado como el que se discute debe corresponder a los códigos hexadecimales 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, F en código ASCII pero el principal problema es que no es así como puede verse en la tabla 3.4 .

Código para cada Tecla asignada como :	Código entregado	Código ASCII que le corresponde
0	11h	30h
1	12h	31h
2	14h	32h
3	18h	33h
4	21h	34h
5	22h	35h
6	24h	36h
7	28h	37h
8	41h	38h
9	42h	39h
A	44h	41h
B	48h	42h
C	81h	43h
D	82h	44h
E	84h	45h
F	88h	46h

Tabla 3.4

Es necesario entonces crear un programa para esta interface:

Para hacerlo totalmente útil el programa debe poder ser llamado en cualquier momento y que pueda ser abandonarlo para que el  $\mu P$  pueda atender otras tareas .

Mantener al  $\mu P$  haciendo un muestreo del puerto 00h (puerto A del PPI ) es un desperdicio y deberá evitarse en el futuro ya que la función principal del  $\mu P$  es el muestreo de señales para la balanza , no el teclado .Una forma de llamado por puerto totalmente independiente del trabajo desarrollado por el  $\mu P$  es el uso de interrupciones ; su forma de atenderlas y el controlador que las atiende y codifica es tema del siguiente apartado .

### 3.9.- INSTALACION DEL SISTEMA DE INTERRUPCIONES

Hasta esta etapa el sistema que se ha conjuntado puede realizar tareas variadas y propias de un circuito basado en un  $\mu P$ . Aún en este estado puede controlar tanto la pantalla y el teclado como la etapa de conversión digital , operando así de manera autónoma .

Para justificar la inclusión de la etapa de interrupciones se cita lo siguiente :

1. Al trabajar con un  $\mu P$  que tiene un servicio de interrupciones que se caracterizan en poder ser llamadas por hardware o software indistintamente y ser atendidas de la misma forma , no sería sensato desperdiciarlas , tomando en cuenta además que este  $\mu P$  puede tener un mapa (área de memoria establecido) de interrupciones desde 0 a 255 "tipos" de interrupciones a las cuales puede servir .

2. Al utilizar el servicio de interrupciones del  $\mu P$  se evita trabajo excesivo que disminuye la velocidad del mismo .

3. Cuando se usan las interrupciones para control del hardware externo se evitan problemas graves de sincronización dentro de programas ejecutados con funciones distintas .

4. Debido a la forma de operación del convertidor que se desea utilizar, operar el programa principal que es el programa de conversión analógico digital mediante interrupciones, da la

oportunidad de crear este programa para una operación más exacta y menos crítica. De esta manera el  $\mu P$  puede realizar tareas variadas mientras el circuito no opere como balanza electrónica.

Se debe poner énfasis al llegar a este punto en dos aspectos básicos de la instalación de un sistema controlador de interrupciones : la instalación hardware y la programación del equipo, o la forma de atender a las interrupciones que se generen.

Una de las particularidades del sistema que se utilizará es que puede reprogramarse una vez que ya este funcionando diferenciándose de otros periféricos programables cuya programación se realiza una sola vez al activar su funcionamiento .

Las interrupciones son fundamentalmente un salto automático a una rutina en software. El salto puede ser ocasionado tanto por software como por hardware debido a la arquitectura interna del 8088. Este  $\mu P$  como otros de la familia Intel , tienen un extenso repertorio de instrucciones abocadas por completo a la atención de interrupciones. Para hacer el manejo de interrupciones lo más flexible posible , hay tres tipos de interrupciones alambradas dentro del 8088 y como estas tienen algunas cosas en común es mejor desglosar su funcionamiento para poder utilizarlas correctamente y entender las diferencias entre ellas .

El 8088 puede manejar un total de 256 interrupciones. Algunas generadas automaticamente por el  $\mu P$ , (tales como la "División entre Cero" o una "Interrupción por Desbordamiento"). Algunas solo son generadas por software (por el uso de la instrucción "INT") y algunas son generadas por hardware (usando las terminales INTR o NMI del 8088). Cada una de estas tres clases de interrupciones tienen características unicas propias pero todas se manejan como saltos a subrutinas en memoria.

Las direcciones de inicio de todas las rutinas de manejo de subrutinas estan localizadas en una tabla que se encuentra en la parte baja del mapa de memoria, desde la 00000h hasta la 003FFh para ser exacto. La razon para utilizar tanto espacio de memoria para las direcciones de 256 interrupciones (se emplazan en una area de 1024 bytes) es que cada dirección tiene 4 bytes de longitud. Los primeros 2 bytes son el offset de la dirección y los segundos 2 bytes son la dirección del segmento.

Toda interrupción dentro del esquema del 8088 tiene un número desde 0 a 255. Cuando una interrupción es generada (ya sea por hardware o por software) el 8088 hace la búsqueda de la dirección correspondiente en la tabla y salta inmediatamente a la rutina almacenada en tal localidad. Partiendo de que la tabla esta localizada en RAM, esta tiene que ser construida por el programador y si un sistema se diseña para utilizar interrupciones como lo es esta tesis, lo primero que se tiene

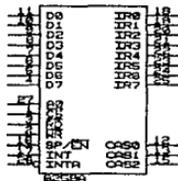
que hacer es construir la tabla para el número de interrupciones (llamados "tipos de interrupciones") que se piensen utilizar .

Uno de los primeros pasos que deben darse para instalar un sistema como este es la forma en que funciona el equipo que estará dedicado al servicio , vectorización de interrupciones y de generación de las señales que activan la capacidad del 8088 para ejecutar llamadas a procesos invocados vía las interrupciones. Se mencionará de manera muy general como funciona el controlador capaz de integrar en un solo circuito todas las características necesarias para el manejo de interrupciones .

#### 3.10- EL CONTROLADOR DE INTERRUPTIONES PROGRAMABLE 8259A.

El controlador 8259 (figura 3.10) es una circuito que puede controlar hasta 8 líneas de interrupción y decir cual es la más importante. El PIC tiene por característica primordial que se puede programar su funcionamiento , esto quiere decir , que debe inicializarse cuando comienza a trabajar y que también puede modificar su operación cuando ya este funcionando .

Para explicar su funcionamiento es necesario conocer la labor que desempeñan cada una de las terminales del conjunto. Para C.I. de complejidad similar y mayor que el PIC sus terminales se clasifican típicamente en cuatro grupos: potencia ó suministro de voltaje , control , datos y direcciones .



NUMEROS DE LAS TERMINALES DEL 8259

D7-D0	BUS DE DATOS BIDIRECCIONAL
IR0	ENTRADA DE LECTURA
IR1	ENTRADA DE ESCRITURA
IR2	ORDEN DE SELECCION DE DIRECCION
IR3	SELECCION DE CHIP
CAS 2, CAS 0	LINEAS CASCADEA
SP/EN	PUERTA DISPONIBLE PROGRAMA ESCLAVO
INT	SEALIDA INTERRUCCION
INTR	ENTRADA RECONOCIMIENTO DE INTERRUCCION
INTA-INTZ	ENTRADAS PETICION INTERRUCCION

FIGURA 3.10: CONTROLADOR PROGRAMABLE DE INTERRUCCIONES 8259

#### Grupo de potencia.

terminal 14 GND y terminal 28 VCC. Tierra y voltaje respectivamente, pueden trabajar a niveles TTL GND=0 , VCC=5±10% .

#### Grupo de control.

Terminal 17 INT (Interrupción) y terminal 26 INTA (Reconocimiento de Interrupción ). Son conectadas directamente al 8088 y obedecen exatamente en niveles y lapsos de tiempo a los requerimientos y secuencia de operación de interrupciones del 8088. Son conectadas a las líneas con los mismos nombres en el 8088 y este, espera una señal de 0 lógico en su terminal INT para comenzar el esquema de interrupciones .

Terminales 1  $\overline{CS}$  , 2  $\overline{WR}$  v 3  $\overline{RD}$ . Estas líneas desempeñan las mismas funciones que en otros controladores programables :  $\overline{CS}$  es activado a 0 cuando se desea hacer programación ó inicialización del 8259, ó bien que permanezca funcionando en el sistema. Así , activando a  $\overline{WR}$  a 0 se posibilita la escritura de comandos y datos en el 8259 y finalmente la terminal  $\overline{RD}$  es usada para que el PIC envíe los datos correspondientes al tipo de interrupción de que se trate .

Terminales 12 CAS0 , 13 CAS1 y 15 CAS2. Estas terminales son utilizadas cuando se debe manejar más de 8 tipos de interrupciones creando un bus privado de interrupciones. Para

que 8259 que funcione como "amo" estas terminales funcionan como salidas y le permiten acceder por sus terminales de interrupción, 8 PIC "esclavos" que se comunican con el CI maestro teniendo sus terminales CAS como entradas, pudiendo así construir un sistema de interrupciones de hasta 64 niveles de servicio.

Terminal 16  $\overline{SP/EN}$ . Es usada en sistemas con múltiples 8259 la cual permite saber si el sistema está operando como "amo" con la terminal  $SP/EN = 1$  lógico, o como "esclavo" con la terminal  $SP/EN = 0$  lógico. Puede ser utilizada como un habilitador de bus al conectarse a su bus de datos un buffer octal, debido a que el 8259 cada vez que envíe información pondrá un 0 en la terminal  $\overline{SP/EN}$ .

#### Grupo de transferencia de datos.

El primer grupo de dos buses es un bus de datos llamado Bus de Petición de Interrupción (terminal 18 a la terminal 25) al que están conectados aparatos externos, los cuales al requerir servicio ponen una señal alta en la terminal que los sirva. A diferencia de otros sistemas que sirven a la primera señal que aparezca en un bus similar el 8259 sirve a la que le haya sido programada como de mayor prioridad, por lo que se conoce como un controlador prioritario. Es una de sus características que durante la programación pueda definirse la jerarquía de operación y que incluso, siguiéndola el 8259,

pueda "interrumpir" el servicio de una interrupción para dar atención a otra de mayor peso dentro del sistema .

El segundo bus de datos son las 8 terminales bidireccionales que son usadas para transferir datos entre el controlador y el sistema, usadas en conjunción con sus terminales  $\overline{CS}$   $\overline{WR}$  y  $\overline{RD}$  . Aquí pueden viajar tanto los datos de programación del PIC como la información de vectorización producida por el 8259 cuando sirve a una interrupción .

#### Grupo de direcciones.

El bus de direcciones del PIC se compone de una sola línea de ancho , A0 , la cual se conecta directamente al bus de direcciones del sistema , operando en alto para algunas palabras de control y en bajo para cuando se envien otras ; esta línea no es estrictamente una línea de direcciones ya que actúa por sí sola como una línea de comando para lo que podría definirse como el selector de dos registros de instrucciones distintos .

#### Programando el 8259A

Como el 8088, el 8259 puede considerarse como un pequeño  $\mu P$  a partir de que puede ser programado para que trabaje de distintas formas. También puede aplicarse el mismo criterio al 8255, pero es más versátil e "inteligente" el 8259 y como es de esperarse , es más difícil de programar .

Hay dos tipos de instrucciones que pueden enviarse al 8259. Las primeras son las palabras de comando de inicialización ó ICW (Initialisation Command Words). Estas deben ser enviadas antes de que exista petición alguna de interrupción. Una vez que el C.I. sea inicializado puede aceptar el envío de las palabras operacionales de comando ó OCW (Operational Command Words). Estas le dicen al 8259 como priorizar las peticiones de interrupciones , si debe ó no ignorar por completo algunas peticiones , poner el estatus actualizado en el bus de datos del sistema y otras funciones más (figura 3.11).

Se requiere de cuatro palabras de inicialización para definir los parámetros básicos de operación para el 8259 y una vez que han sido enviados al C.I. ( usualmente al inicio del programa ) ya no es necesario ,por lo general , tratar con este circuito. Desde que las ICW son el equivalente al hardware de control de este circuito debe tenerse especial cuidado de enviarlas apropiadamente .

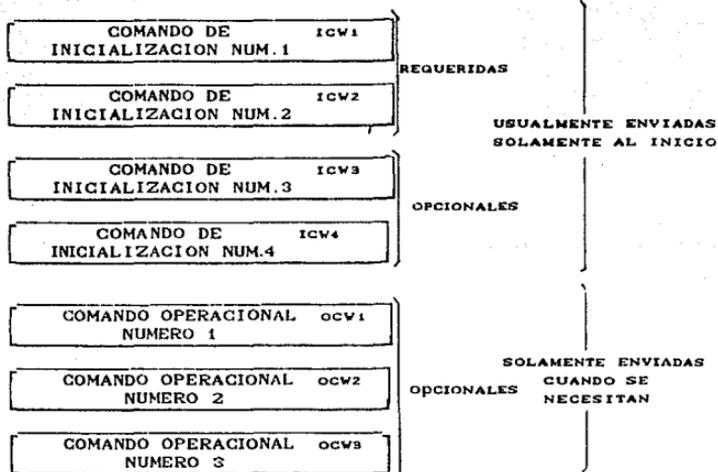


Fig 3.11 PARAMETROS DE PROGRAMACION PARA EL 825P

Cada una de las cuatro ICW establecon diferentes parametros , así que, una inicialización típica no solo utiliza diferentes configuraciones para cada ICW sino que utiliza también un número diferente de ICW. Donde un sistema completa su inicialización con una ICW1 y una ICW2 otro puede necesitar de las ICW1 la ICW2 y la ICW4. Para una información completamente detallada referirse a las hojas de datos del manual de microprocesadores de Intel 1991 en el apéndice.

A continuación se ofrece un resumen condensado del significado de cada bit para cada una de las palabras de

inicialización que se utilizan para un sistema que tenga un solo 8259 trabajando de manera autónoma y como único controlador de interrupciones de dicho sistema.

Es obvio que la primera palabra que debe ser enviada debe ser la ICW1, la cual esta compuesta por 8 bits y necesita de la presencia de A0 la cual lleva la siguiente información:

**PALABRA DE INICIALIZACION 1 (ICW1).**

**A0** Utilizada como conmutador entre dos registros de instrucciones diferentes dentro del 8259. Su función puede visualizarse de la siguiente manera:

**PUERTOS PARES**, con A0 en alto (A0=1)

**PUERTOS NONES**, con A0 en bajo (A0=0). Para transmisión específica de las ICW2 y la ICW4.

**D0** 0

1 Indica que el usuario enviará la ICW4

**D1** 0 Listo para operar en la modalidad master-slave.

En este caso es necesario preparar el envío de la ICW3.

1 Solo existe un 8259 operando en los buses del sistema.

**D2** 0 Modalidad de 8 bytes , especial para trabajar en un sistema con el 8085 como  $\mu P$  en el sistema.

1 Modalidad de 4 bytes para el 8088 ,es decir , si debe estar en comunicación con un 8088.

**D3** 0 Modalidad de "disparo por flanco".La transición de un estado bajo a uno alto en cualquiera de las terminales de petición de interrupción dispararán al 8259.

D4	0	Para aplicaciones que utilicen el 8085.
	1	Debe ser enviado en conjunción con A0=0 para indicar la transmisión de la ICW tipo 1.
D5	0	Solamente son necesarios en la modalidad 8080 ó
	1	8085.
D6	0	
	1	
D7	0	
	1	

#### PALABRA DE INICIALIZACION 2 (ICW2).

La ICW2 provee los datos para situar la tabla de vectores de interrupciones dentro del mapa de memoria total del sistema .

D0	0	
	1	Los primeros tres bits son utilizados para apuntar
D1	0	al vector de interrupción el cual se desee establecer
	1	son útiles unicamente en la comunicación con $\mu$ P 8080
D2	0	y 8085 .
	1	

D3	0	
	1	Los últimos 5 bits corresponden a los 5 bits de
D4	0	mayor orden del tipo de interrupción. Los tres más
	1	bajos dependen, de cual de las líneas de
D5	0	interrupción fue activada.
	1	
D6	0	
	1	
D7	0	
	1	

La forma en que esta palabra de control se utilice dentro de este sistema específico se discutirá en la sección de emplazamiento de la tabla de vectores dentro del mapa general de memoria .

#### PALABRA DE INICIALIZACION 3 (ICW3).

Debido a que dentro del sistema no existe más que un 8259 esta palabra de control no es necesario enviarla .

#### PALABRA DE INICIALIZACION 4 (ICW4).

**A0** Utilizada para indicar que se envía o una ICW2 o una ICW4.

**D0** 0 Indica que el controlador esta conectado a un 8085.

1 Indica que el controlador esta conectado a un 8088.

**D1** 0 Con este bit en cero lo que el controlador espera es

una instrucción de fin de interrupción (IRET). Pero existe como desventaja que debe vigilarse y checar más software .

1 Para dar fin automáticamente a una interrupción y quedar en posibilidad de responder una nueva interrupción, justo después del último pulso de reconocimiento de interrupción enviado por el 8088 via la terminal INTA (que también así es llamada en el 8259).

D2 0 Establece que la modalidad de buffer está desactivada. Esto quiere decir que por la terminal  $\overline{SP/EN}$  no se enviará ningún pulso para habilitar buffers que separen al sistema del controlador.

1 Habilita la modalidad de buffer. Cuando sea necesario enviar el vector de la interrupción al  $\mu P$ , la terminal  $\overline{SP/EN}$  puede ser utilizada para habilitar buffers que den mayor calidad a la señal y aislen cada sección de manera segura en caso de un accidente

D3 0 Identifica al 8259 que se programe como un esclavo.

1 Le da el status de controlador maestro de todo el sistema de interrupciones

D4 Es usado para activar ó desactivar la modalidad especial de red total ó "Fully Nested Mode".Util solo en redes con varios 8259

0 Deshabilita esta modalidad .

1 La habilita .

Los tres últimos bits deben ponerse a cero , debido a que constituyen la firma distintiva , junto con A0 , de la ICW1 .

D5 0

D6 0

D7 0

#### Programación dinámica del 8259 .

Una vez que las ICW han sido enviadas al 8259 , el C.I. esta listo para comenzar a vigilar las peticiones de interrupciones. Las palabras de mando operacionales no son indispensables para que el circuito opere , las ICW si son comandos necesarios , el integrado no hará nada hasta que estas sean enviadas. Por otra parte las OCW son comandos opcionales que permiten tanto modificar el desempeño del circuito como permitir que el 8088 lea los registros internos del 8259. A partir de que estas son opcionales , la mayoría de las aplicaciones que las usen las enviarán en base a necesidades que deben ejecutarse durante la operación del circuito.

#### PALABRA OPERACIONAL 1 (OCW1).

Esta palabra de control es utilizada para mascarar las interrupciones que se deseen no sean habilitadas por accidente o señales de ruido en el circuito. Cada uno de sus 8 bits habilita o deshabilita la operación de cada una de las 8 interrupciones que el 8259 vigila:

0 Habilita la interrupción .

1 Mascara la interrupción .

#### PALABRA OPERACIONAL 2 (OCW2).

D0 Estos tres bits establecen la prioridad de las

D1 interrupciones 000 es la posición de default, dando a IR0

D2 la más alta prioridad y a IR7 la prioridad menor.

El uso de estos tres bits "desplaza la prioridad" de derecha a izquierda . comenzando con la prioridad mas alta para la linea que sigue a la que es escogida como de prioridad más baja. Esto se explica como sigue:

	IR7	IR6	IR5	IR4	IR3	IR2	IR1	IR0	Interrupciones
000	8	7	6	5	4	3	2	1	Por default .
010	5	4	3	2	1	8	7	6	El bit que fué escogido es el correspondiente a la IR2 , dandole la prioridad mas baja 8 , siendo ahora establecido como el de mayor peso el bit que corresponde a la IR3 .

Para que esta nueva jerarquización sea obedecida es necesario que en la OCW3 el bit 6 sea puesto a 1.

D3 0 Deben ser puestos a cero porque constituyen la firma

D4 0 distintiva de la OCW2 .

D5 0 Este bit prepara al 8259 para que ignore una instrucción de fin de interrupción.

1 Hace que el controlador obedezca una instrucción de fin de interrupción .

- D6** 0 Ignora lo que se ha establecido en los bits D0 , D1, D2 , D3 , D4 , D5 y D7 .
- 1 Habilita ó confirma lo que esta siendo establecido es los bits D0 , D1 , D2 , D3 , D4 , D5 y D7 .
- D7** Puede considerarse como una bandera de seguridad , como se verá:
- 0 Las condiciones de default se mantienen.
- 1 En combinación con los otros bits regula la acción del 8259 cuando una orden de fin de interrupción es recibida y ejecutada. Las combinaciones de bits que se reciban diferentes a las condiciones de default serán respetadas

#### PALABRA OPERACIONAL 3 (OCW3) .

- D0** Son usados para leer los registros internos del 8259 .
- D1** Dependiendo de como se pongan estos bits se tiene: que interrupción es leída :llamando al ISR ó Registro de servicio activo (In service register), ó que interrupción está en espera de servicio: llamando al registro de petición de interrupción .
- D2** Este es activado para realizar un "rastreo" ó "polling mode". Cuando es activado puede utilizarse el software para checar el estado del controlador actuando de manera muy próxima como lo hace un ciclo de INTA pero en lugar de regresar el valor de un

vector de interrupción pondrá en el bus de datos una "palabra de rastreo" o "polling word" de la cual solo importan los bits D0, D1, D2, D3 y D7 con estas funciones:

- D7 0 La línea de petición de interrupción ha sido activada.
- 1 La línea de petición de interrupción debe ser atendida.
- D0 Dan el número de interrupción que ha sido activada.
- D1
- D2
- D3
- D3 0 Da la identificación de la OCW3.
- D4 1
- D5 Usados para definir el modo especial de máscara.
- D6 "Special Mask Mode". Cuando surge la necesidad de reconocer y servir una interrupción de menor prioridad de la que se está sirviendo y se activa esta modalidad si se sirve a todas las interrupciones de jerarquía mayor o menor. Exceptuando a aquella que se este sirviendo y a las que hayan sido deshabilitadas por la ICW4. En otras circunstancias detiene la ejecución de la rutina de servicio y atiende la nueva petición.
- D7 Se trata de un bit de seguridad, con el cual se toma en cuenta o no lo que se establece en el bit 6. Para esto es necesario ponerlo a 1 lógico.

### 3.11- MAPA DE MEMORIA DEL SISTEMA Y LA UBICACION EN EL MISMO, DE LOS VECTORES DE INTERRUPCIONES

Para poder utilizar el servicio de interrupciones es necesario definir como se va situar tanto el controlador dentro del mapa de puertos y más importante aun , dentro del mapa de memoria donde van a emplazarse los vectores de atención a subrutinas .

#### Mapeo de los vectores de interrupción.

Para un sistema de control del tamaño tan reducido como el que se diseña , la parte más difícil de mantener es el mapa de memoria disponible , ya que la tendencia es que sea lo más pequeña posible y así, que el sistema permanezca lo más autónomo posible respecto a requerimientos de almacenamiento .

La compañía que produce tanto al  $\mu P$  como al controlador programable de interrupciones ha establecido una área de memoria muy grande reservada para vectores de interrupción ya establecidos así como para vectores utilizados por equipo que ellos produzcan en el futuro. Esta área de memoria contiene las direcciones para 31 vectores de interrupción reservados y se sabe que cada vector es formado por 4 bytes de dirección , con los cuales se ocupa 124 bytes de la parte baja de la memoria RAM para no dejar tanto espacio libre dentro de un circuito que no va ha ser ampliado con hardware que esta dedicado para este sistema (figura 3.12).

Las interrupciones que serán respetadas para el uso que fuerón inicialmente creadas son las primeras 4 , que son utilizadas exclusivamente por el  $\mu P$ , para reglamentar su funcionamiento interno. Estas interrupciones están alambradas internamente. A partir de la interrupción número 4 Intel establece las direcciones de vectores para futuras expansiones para este  $\mu P$  ó para otros más avanzados de la misma familia .

Para este proyecto solo se respetan tres de estos vectores reservados para el caso de que se produzca, en un futuro , hardware que sea útil al integrarlo a este diseño para aumentar la capacidad ó versátilidad de la balanza .

A partir de la interrupción "tipo 8" se emplazaran las direcciones de servicio a las rutinas especiales creadas para servicio a la balanza ; como un solo controlador 8259a puede acceder 8 diferentes interrupciones , se incluirá en el mapa de este proyecto hasta la interrupción "tipo F" .

MAPA DE INTERRUPCIONES DEL SISTEMA.

00040h	INTERRUPCION F	USADO POR HARDWARE EXTERNO DE INTERRUPCION
0003Ch	NUMERO 8 DISPONIBLE	
	INTERRUPCION E	
00038h	NUMERO 7 DISPONIBLE	
	INTERRUPCION D	
00034h	NUMERO 6 DISPONIBLE	
	INTERRUPCION C	
00030h	NUMERO 5 DISPONIBLE	
	INTERRUPCION B	
0002Ch	NUMERO 4 DISPONIBLE	
	INTERRUPCION A	
00028h	NUMERO 3 DISPONIBLE	
	INTERRUPCION 9	
00024h	NUMERO 2 DISPONIBLE	
	INTERRUPCION 8	
00020h	NUMERO 1 DISPONIBLE	RESERVADO POR INTEL
	INTERRUPCION 7	
0001Ch	RESERVADA	
	INTERRUPCION 6	USADO POR EL 8088
00018h	RESERVADA	
	INTERRUPCION 5	
00014h	RESERVADA	
	INTERRUPCION 4	
00010h	SOBREFLUJO	
	INTERRUPCION 3	
0000Ch	TIPO DE UN SOLO BYTE	
	INTERRUPCION 2	
00008h	NMI	
	INTERRUPCION 1	
00004h	UN SOLO PASO	
	INTERRUPCION 0	
00000h	DIVISION ENTRE CERO	

Figura 3.12

#### Mapeo del 8259

Si se toma en cuenta que el 8259 es un circuito programable deberá entonces establecerse dos puertos dentro del mapa de puertos para que ahí pueda ser llamado mediante instrucciones IN y OUT de parte del 8088.

Como es más sencillo el direccionamiento estático que el dinámico se prefiere direccionarlo en puertos del 00h al FFh (0 a 255). Los puertos 00h al 03h son dedicados al PPI(A), más sin embargo la configuración hardware puede expandirse a otro PPI para acceso a puertos del 04h al 07h. Se pueden utilizar los puertos que le siguen, pero existe una diferencia con los puertos ya establecidos. Estos puertos son de operación específica a diferencia de los anteriores que son de propósito general.

Como se hizo con la memoria RAM y la ROM se desea también dividir el área de puertos solo para simplificar y ordenar que salidas y entradas del sistema serán consideradas de uso general y cuales de uso específico. Por lo tanto se divide en dos exactamente este espacio. Para realizar esta disposición se puede utilizar una señal de dirección como conmutador, A7, con la que se separan los puertos de propósito general 00h al 07Fh con A7=0 y del puerto 80h al 0FFh con A7=1. Así no se tendrán problemas al codificar el puerto 08h y 09h que pueden translapar condiciones con el puerto 00h y 01h.

Los puertos para el 8259 son entonces el 80h y el 81h

DIRECCION DE PUERTO			
	PUERTO 0FFFh	DIRECCIONADOS A TRAVES DE DX	
	PUERTO 0FFEh		
	”		
NO SON USADOS	PUERTO 0FFh	SELECCIONADOS CON A7 A 1 LOGICO	
	PUERTO 0FEh		
	PUERTO 0FDh		
	PUERTO 0FCh		
	PUERTO 0FBh		
	”		
PUERTOS DEL 8259A	PUERTO 0B1h		
	PUERTO 0B0h		
	”		
NO USADOS	PUERTO 07Fh	SELECCIONADOS CON A7 EN 0 LOGICO	
	PUERTO 07Eh		
	PUERTO 07Dh		
	”		
	”		
PUERTOS DEL 8255(A)	PUERTO 03h		
	PUERTO 02h		
	PUERTO 01h		
	PUERTO 00h		

Figura 3.13. MAPEO DE PUERTOS PARA EL SISTEMA CONTROLADOR.

### Instalación del hardware.

Para una posible instalación práctica del sistema es necesario :

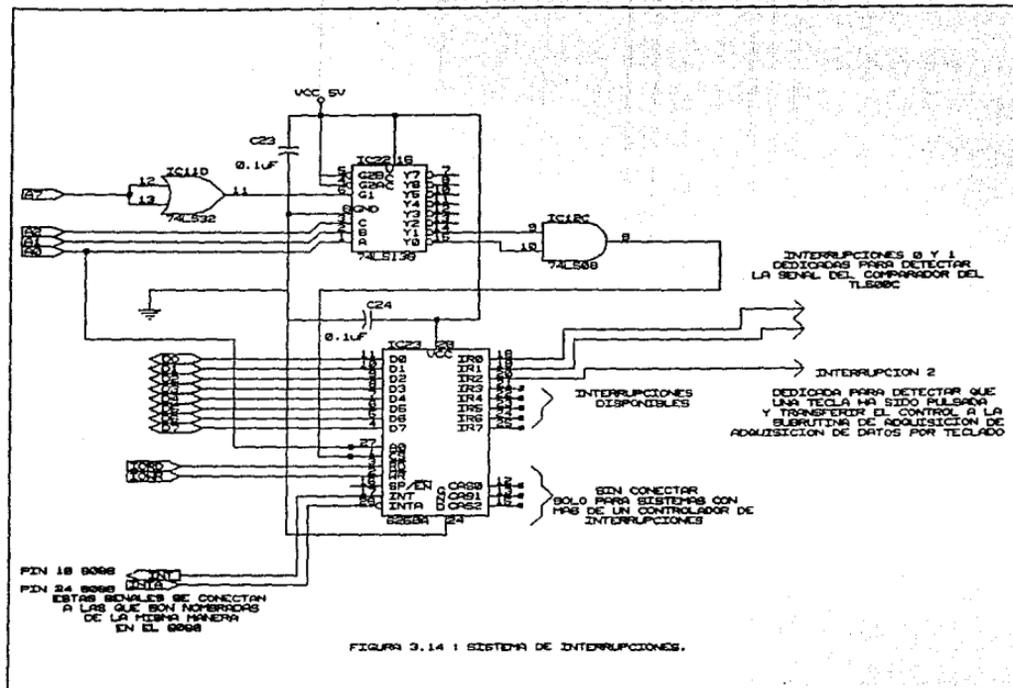
1	8259
1	74LS138
2	Cap. 0.1 $\mu$ F
1	Cap. 68pF

Debera conectarse el circuito de la figura 3.14 :

La compuerta LS32 solo sirve como buffer que acople la señal A7 con el tercer selector del 74LS138. Al mismo se le conectan , para decodificar los puertos : A0 , A1 , A2 para escoger 1 de 8 puertos posibles en las posiciones 80h, 81h, hasta el 87h. El LS08 permite que cuando se seleccionen los puertos 80h y 81h se garantice habilitar a  $\overline{CS}$  del 8259 con una señal de 0 lógico .

A0 , como ya se vió en la sección de programación , sirve para elegir dos registros de comando diferentes para las ICW y las OCW .

Deben conectarse al 8088 las señales INTA e INT y de la lógica de puertos se conectan a  $\overline{RD}$  ,  $\overline{IORD}$  y a  $\overline{WR}$   $\overline{IOWR}$  .



El bus de datos del 8259 va al bus de datos del sistema .  
Quedan listas para recibir señal para interrupción las terminales 18 a 25 y sin conexión quedan las señales que conforman el bus de interrupciones, unico para sistemas con más de un 8259 trabajando .

Como la línea de "cualquier tecla orpimida" se activa a 1 lógico es la señal idonea para activar una interrupción. Además hay que decidir que interrupción va a usarse y como es la primera en ser implementada, estará en la terminal 18 o IR0.

Las interrupciones pueden manejar las requisiciones de atención de casi cualquier periférico que se quiera controlar ya que lo unico que se necesita es una señal de nivel alto TTL y programación de los manejadores adecuados de hardware .

Hasta aqui se ha cubierto la mitad del planteamiento sobre manejo de estas estructuras ya que es necesario construir la tabla de vectores , que además puede ser utilizada mediante instrucciones de software INT.

La forma exacta de como una petición se activa es la siguiente:

Tan pronto como el 8259 ve un 1 lógico en cualquiera de sus líneas IR , deja saber al 8088 de esto enviando una señal desde su terminal INT a la terminal INTR del 8088. Cuando el  $\mu P$

recibe la petición de interrupción . pone dos ciclos de reconocimiento de interrupción. Estos son similares a los ciclos de bus pero hay dos pulsos de reloj insertados entre ellos. El 8088 deja saber al 8259 que ha recibido una señal de 0 durante el tiempo T2 y T3 del primer ciclo de bus para reconocimiento de interrupción .

En el siguiente ciclo de bus . el 8088 articulará un 0 durante el tiempo T2 y T3 y esa es la ocasión en que el 8259 tiene que poner en el bus de datos el vector activado. El 8088 tomará el tipo de interrupción . lo multiplica por 4 para tener la dirección de la localidad correspondiente en la tabla de vectores de interrupciones y entonces ejecuta la rutina almacenada en tal dirección .

## C A P I T U L O   I V

### 4.1.-ETAPA DE AMPLIFICACION Y FILTRADO DE LA SENAL

El amplificador de instrumentacion tiene como finalidad amplificar la señal proveniente de un transductor dado que, dicha señal, es apenas de unos cuantos milivolts.

Una vez amplificada la señal a una escala conveniente se puede interpretar el comportamiento del fenómeno estudiado.

#### Diseño y construcción del amplificador de instrumentación.

El amplificador de instrumentacion consta de varios amplificadores operacionales (AO), que acoplados en configuraciones basicas amplifican la señal deseada.

Pero ¿Que es un amplificador operacional?

Un amplificador operacional (AO) es un amplificador de alta ganancia, acoplado directamente, al que se le agrega una realimentacion para controlar la caracteristica de respuesta. Se emplea para realizar una amplia variedad de funciones lineales (y tambien algunas no lineales) y normalmente se conoce por el nombre de circuito integrado lineal basico (o mas comunmente analogico).

El amplificador operacional ideal tiene las siguientes caracteristicas :

- a) Resistencia de entrada  $R_i = \infty$
- b) Resistencia de salida  $R_o = 0$

- c) Ganancia de tensión  $A_v = -\infty$
- d) Ancho de banda = infinito
- e) Equilibrio perfecto  $V_o = 0$  cuando  $V_x = V_y$
- f) Ausencia de desviación en las características con la temperatura.

El amplificador de instrumentación fue armado con CI's LM741.

La figura 4.1 muestra el circuito amplificador y el filtro utilizado en este proyecto.

Como se puede observar el circuito consta de cinco etapas:

- I) Seguidor de voltaje
- II) Amplificador diferencial
- III) Amplificador Inversor de ganancia ajustable
- IV) Compensación de offset.
- V) Filtro pasa-bajas

Cada una de las etapas se describen a continuación:

Al observar la figura 4.2a, se detecta que hay un divisor de voltajes formado con el puente de las galgas y la resistencia  $R=100K$  del amplificador diferencial ocasionando que se tenga un voltaje falso proporcionado por el transductor.

Para evitar esto se colocan dos seguidores de voltaje mostrados en las figuras 4.1 y 4.2b. Como se puede observar los seguidores de voltaje separan al puente de galgas de la etapa de amplificación. La ecuación matemática para el seguidor de

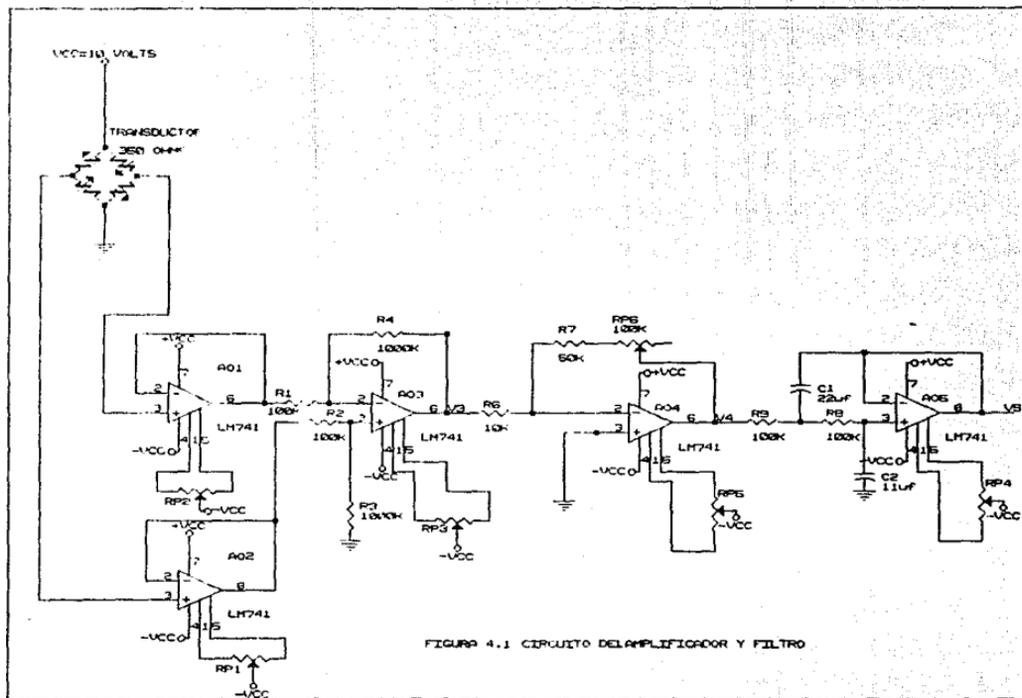


FIGURA 4.2A

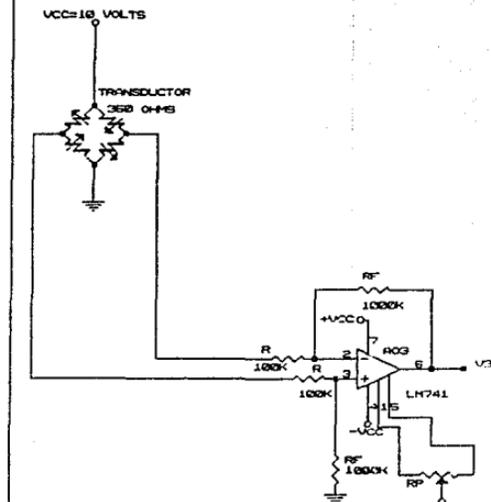
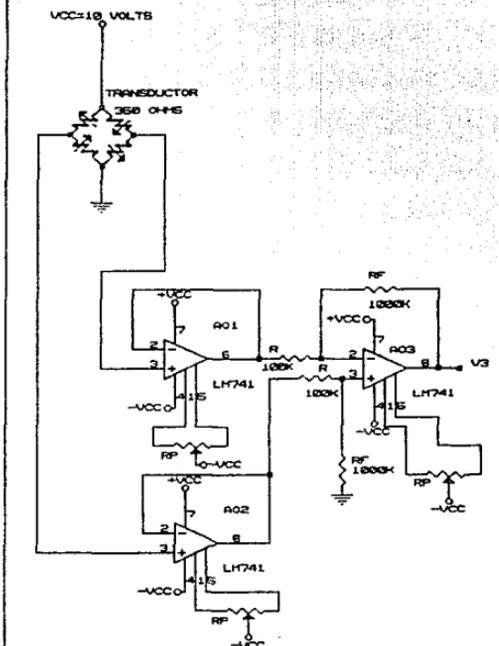


FIGURA 4.2B



voltaje de la figura 4.3a es :

$$V_1(t) = V_0(t)$$

Los seguidores de voltaje van acoplados a las entradas de un amplificador diferencial. Este amplificador tiene como función hacer la resta o diferencia de los voltajes de los seguidores ( $V_1$  y  $V_2$ ) ver la figura 4.3b. El voltaje resultante  $V_3$  es la señal que nos representa la diferencia de potencial que presentan las galgas al ser sometidas a una deformación.

El voltaje obtenido en el puente de las galgas es de unos cuantos milivolts (aproximadamente 38mV a plena escala) por lo que, en esta etapa, la señal recibe el primer paso de amplificación, ya que no se puede amplificar la señal al valor deseado porque el amplificador muestra inestabilidad. La ecuación matemática es :

$$V_3 = R_f \cdot (V_2 - V_1) / R$$

La señal que sale del amplificador diferencial entra a su vez a un amplificador inversor de ganancia ajustable.

Esta etapa tiene como objetivo dar el segundo paso de amplificación.

La señal es calibrada a una escala conveniente para que el convertidor pueda empezar a funcionar (sin que haya carga en la barra) el convertidor deberá poner a cero los displays y al colocar una carga de 200 gramos, el potenciómetro debe ser ajustado para obtener una lectura correcta.

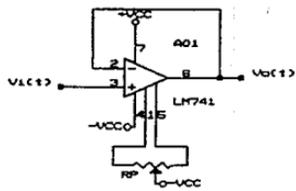


FIGURA 4.3A  $V1(t) = V0(t)$

SEGUIDOR DE VOLTAJE

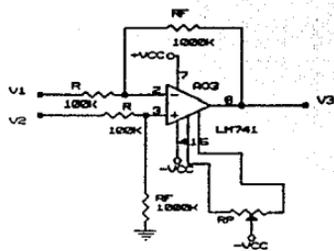


FIGURA 4.3B  $V3 = RF \cdot (V2 - V1) / R$

AMPLIFICADOR DIFERENCIAL

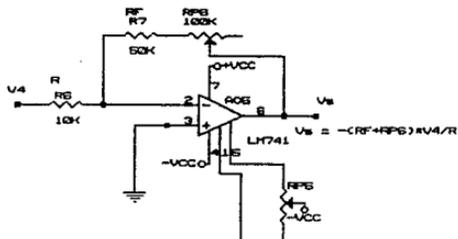


FIGURA 4.3C

AMPLIFICADOR DE GANANCIA AJUSTABLE

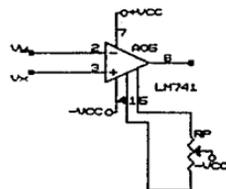


FIGURA 4.3D

TECNICA UNIVERSAL DE BALANCEO

En la figura 4.3c se muestra la configuración básica de este tipo de amplificador.

Este amplificador funciona de la siguiente manera:

$$V_4 = -V_3 \cdot R_f / R$$

Por consiguiente , si el voltaje  $V_3$  es de cierta polaridad el voltaje de salida del amplificador se defasa  $180^\circ$ .

Un amplificador operacional ideal está perfectamente equilibrado , o sea que  $V_o=0$  cuando  $V_x=V_y$ . Un amplificador operacional real tiene cierto desequilibrio provocado por la falta de apareamiento de los transistores de entrada. Este desajuste provoca unas corrientes de polarización desiguales a través de las terminales de entrada y se requiere una tensión de equilibrio entre dichas terminales para tener equilibrada la salida.

Por ello cuando se emplean amplificadores operacionales a menudo se necesita equilibrar la tensión offset de salida. Esto significa que debemos aplicar una pequeña tensión continua en la entrada de tal manera que a la salida la tensión obtenida sea cero, tal como se muestra en la figura 4.3d.

La última etapa del circuito de la figura 4.1 es un filtro pasa-bajas. La señal de salida puede contaminarse con señales de ruido por lo cual es recomendable el uso de un circuito que elimine estas señales parásitas, las cuales tienen una alta frecuencia. Considerando esta característica debe implementarse un filtro pasa-bajas con una frecuencia de corte muy baja, de aproximadamente 0.1 Hz, con lo cual se logra que solo pasen señales de corriente directa.

Para el diseño del filtro butterworth tomando valores no calibrados a escala  $C1''$  y  $C2''$ :

$$C1'' = 1.414F \quad \text{y} \quad C2'' = 0.7071F$$

Calibrando a escala de frecuencia  $C1'$  y  $C2'$  ( $R1=R2$ )

$$C1' = C1'' / (2\pi * 0.1) = 1.414 / (2\pi * 0.1) = 2.2504F$$

$$C2' = C2'' / (2\pi * 0.1) = 0.7071 / (2\pi * 0.1) = 1.1253F$$

Si  $R=100Kohms$  los valores de calibración a escala de impedancia son:

$$C1 = C1' / R = 2.2504 / 100000 = 22.504\mu F$$

$$C2 = C2' / R = 1.1253 / 100000 = 11.253\mu F$$

## 4.2.-CONVERSION ANALOGICA A DIGITAL

### CONVERTIDOR DE INTEGRACION

La conversión Analógica a Digital es la transformación de señales analógicas en forma digital,

#### Convertidor de integración.

Este tipo de convertidores transforma el cociente de voltajes entre la entrada y la referencia en una relación de tiempos. Existen varias formas de convertidores de integración, pero todas se basan en rampas lineales obtenidas de un integrador analógico controlado respectivamente por una y otra señal. El integrador más conocido es el de tipo integración de doble rampa, cuyo diagrama de bloques se muestra en la figura 4.4. La entrada del integrador se conecta alternativamente a tierra, a la señal de entrada o a una señal de referencia mediante un conmutador. La salida del integrador pasa a un comparador y luego al sistema lógico y de sincronización. Este sistema es el que se encarga también de controlar el conmutador de entrada, los biestables de salida, etc.

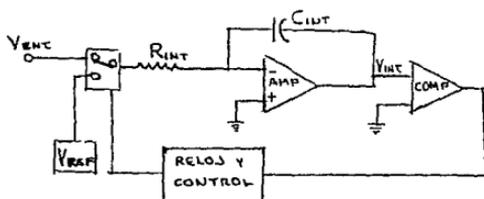


Figura 4.4. Diagrama a bloques del convertidor integrativo básico.

La conversión tiene lugar en tres fases (figura 4.5). La primera fase es la de "autocajuste". Durante ella se ajusta a cero la salida del integrador y por lo general también se anulan los voltajes de desbalance del sistema, por lo que se denomina "fase de autocero". La segunda fase es "integración de la entrada" o simplemente "fase de integración". Durante este período, la entrada del integrador está conmutada a la señal de entrada; por lo tanto, la variable de entrada se integra durante un tiempo fijo, determinado por el sistema de control. La tercera fase recibe el nombre de "integración de referencia" o "fase de desintegración". Durante este período la entrada del integrador se conecta a la referencia durante un tiempo variable, hasta que la salida del integrador regresa a su valor inicial. Y como resultado el tiempo necesario para lograr esto queda registrado en el sistema de control lógico.

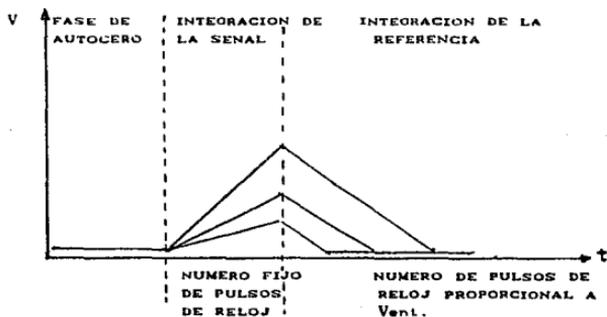


Figura 4.5. Las tres fases de la conversión.

#### 4.3. DESCRIPCION DEL SISTEMA DE CONVERSION.

El procesador analógico TL500C provee las funciones básicas para la conversión analógica a digital por integración de doble rampa.

El TL500C puede ser controlado por el procesador digital TL502C, por lógica discreta ó por una rutina de software de un microprocesador.

El principio básico de conversión por integración de doble rampa es relativamente simple. El capacitor  $C_x$  es cargado por el integrador desde  $V_{ct}$ , durante un tiempo fijo en proporción determinada por el valor del voltaje de entrada desconocido. Entonces el capacitor es descargado en proporción constante (determinado por el voltaje de referencia) de vuelta hasta  $V_{ct}$ , donde el tiempo de descarga es medido con precisión.

Las relaciones del valor de carga y descarga son las siguientes :

$$V_{cx} = V_{ct} - \frac{V_i * T_1}{R_x * C_x} \quad \text{carga} \quad (1)$$

$$V_{ct} = V_{cx} - \frac{V_{ref} * T_2}{R_x * C_x} \quad \text{descarga} \quad (2)$$

Combinando las ecuaciones 1 y 2 resulta:

$$\frac{V_i}{V_{ref}} = - \frac{T_2}{T_1}$$

donde:

$V_{cx}$  = Voltaje de carga a través de  $C_x$  durante  $T_1$  y durante  $T_2$

(igual en magnitud).

$V_{ct}$  = Voltaje inicial del comparador.

$V_1$  = Valor promedio del voltaje de entrada durante  $T_1$ .

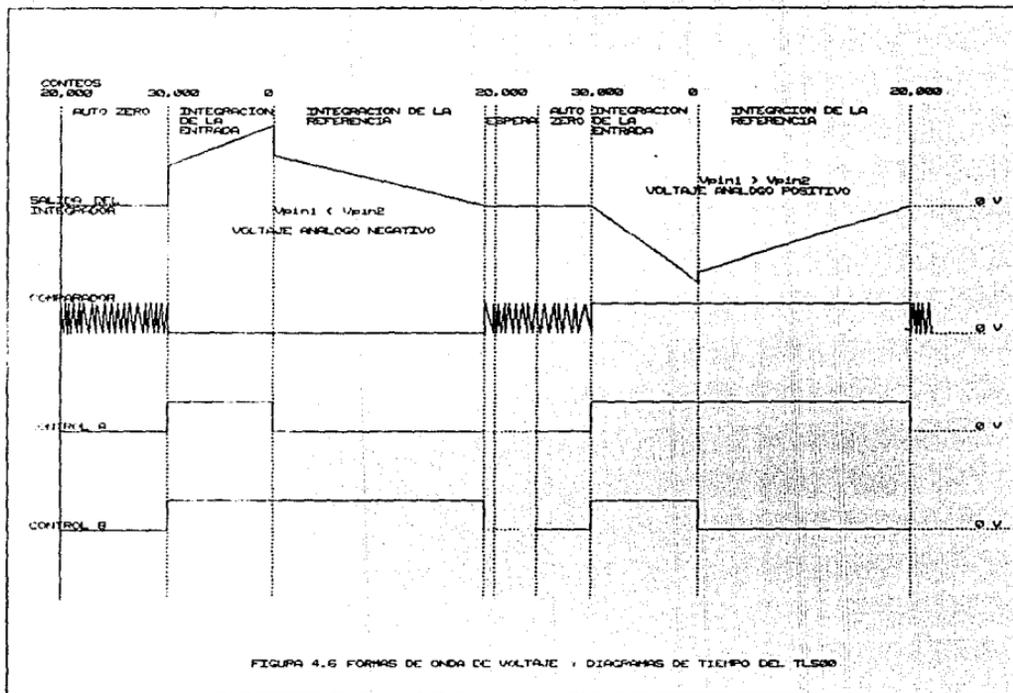
$T_1$  = Período de tiempo en el que el voltaje de entrada desconocido es integrado.

$T_2$  = Período de tiempo en el que el voltaje de referencia es integrado.

#### Fase de autocero.

El ciclo comienza al terminar la fase de integración de la referencia cuando el procesador digital aplica niveles bajos a las entradas A y B del procesador analógico. Las conversiones se ejecutan continuamente. El voltaje de referencia es almacenado en el capacitor de referencia  $C_{REF}$ , el voltaje de offset del comparador es almacenado en el capacitor de integración  $C_x$  y la suma de los voltajes de offset del separador(buffer) y el del integrador son almacenados dentro del capacitor de cero  $C_z$ . Durante la fase de ajuste a cero, la salida del comparador se caracteriza por una oscilación (ciclo límite) con forma y frecuencia onda indeterminada la cual es filtrada y desplaza en el nivel de C.D. por el corredor de nivel(level shifter).Figura 4.6.

TESIS CON  
FALLA DE ORIGEN



#### Fase de integración de la entrada.

La fase de autocero se completa a la cuenta BCD de 30,000 y niveles lógicos altos son aplicados a ambas entradas de control iniciando la fase de integración. El integrador carga Cx para un tiempo fijo de 10,000 cuentas BCD a un rango determinado por el voltaje de entrada. Nótese que durante esta fase, las entradas analógicas solo ven una alta impedancia de las entradas no inversoras de la entrada del amplificador operacional. Por eso, el integrador solo responde a la diferencia de voltajes de las entradas analógicas, contando así con entradas verdaderamente diferenciales.

#### Fase de integración de la referencia.

A la cuenta BCD de  $39,999+1=40,000$  BCD o 0, la fase de entrada a la integración es terminada y se empieza la fase de integración de la referencia muestreando la salida del comparador. Si la salida del comparador es baja, correspondiendo a un voltaje promedio negativo analógico, el procesador digital aplica un nivel bajo y un nivel alto a las salidas A y B, respectivamente, para aplicar el voltaje de referencia almacenado en Cref al separador de impedancias ó buffer. Si la salida del comparador es alta, correspondiendo a una entrada positiva, las entradas A y B se hacen al nivel alto y nivel bajo, respectivamente y el voltaje negativo de referencia almacenado es aplicado al buffer. En ambos casos el procesador selecciona automáticamente el estado lógico apropiado que cause

que integrador integre hacia cero a un rango proporcional al voltaje de referencia. El tiempo requerido para llegar a cero es medido por el contador en el procesador digital. La fase se completa cuando la salida del integrador cruza a cero y el contenido del contador es transferido a registros, o cuando la cuenta BCD del contador alcanza 20,000 y una indicación de sobrerango es activada. Cuando esto ocurre, la señal de sobrerango enciende todos los dígitos mostrando una A en ellos.

#### 4.4. DESCRIPCION DEL PROCESADOR ANALOGICO

El procesador analógico TL500C esta diseñado para una compensación automática interna de offsets, integrar un voltaje de referencia en sentido opuesto y posee una indicación de cruce a cero. El sistema de control externo puede ser un microprocesador y una subrutina de software, lógica discreta ó un controlador TL502C o TL503C. El TL500C esta diseñado para proveer un convertidor de pendiente dual simple, de bajo costo efectivo. El circuito tiene la característica de poseer verdaderas entradas diferenciales, alta impedancia de entrada y una fuente de referencia interna. El TL500C ofrece la posibilidad con una fuente de referencia de precisión externa. Estos circuitos son aptos para operar a un rango de temperatura de 00C a 70C.

## GUIA DE SELECCION DE COMPONENTES EXTERNOS.

El capacitor de autocero  $C_z$  y el capacitor de referencia deben de estar dentro de las condiciones de operación recomendados y deben tener características de baja fuga dieléctrica. La mayoría de los capacitores de dieléctrico de film y algunos capacitores de tantalum dan resultados aceptables. Los capacitores cerámicos y de aluminio no son recomendados por sus características de fuga dieléctrica relativamente altas.

El capacitor de integración  $C_x$  debe estar dentro del rango recomendado y debe tener buena linealidad de voltaje y baja absorción dieléctrica. Un capacitor de dieléctrico de polipropileno se recomienda para una exactitud de  $4\frac{1}{2}$  dígitos. Para aplicaciones de  $3\frac{1}{2}$  dígitos capacitores de policarbonato o de polyster y otros de film dieléctrico son adecuados. Los capacitores cerámicos y electrolíticos no se recomiendan.

El acoplamiento directo de la salida del comparador a cualquier terminal analógica (en orden de importancia 17,18,14,7,6,13,1,2,15) deben minimizarse para evitar oscilaciones. Además en todas las terminales de suministro de voltaje debe aplicarse bypass en la envoltura.

El común digital y el analógico están aislados internamente y deben estar a diferentes potenciales. La tierra digital puede estar entre 4 volts positivos ó negativos del suministro y aun funcionando apropiadamente.

La constante  $R \times C \times$  debe mantenerse cerca dentro de lo posible del valor mínimo dado por la fórmula siguiente:

$$R \times C \times (\text{MINIMA}) = \frac{V_{ID} (\text{full scale})}{V_{OM} - (-V_{I(\text{pin } 2)})}$$

donde:

$V_{ID} (\text{full scale})$  = voltaje en la terminal 1 respecto la terminal 2.

$t_1$  = tiempo de integración de la entrada en segundos.

$V_{I(\text{pin } 2)}$  = voltaje en la terminal 2 respecto de la tierra analógica.

#### 4.5. UNION DEL PROCESADOR ANALOGICO AL SISTEMA

Para acoplar esta etapa es necesario reunir al mismo tiempo tanto los elementos y señales necesarias para hardware así como las rutinas software para operar el convertidor y realizar este proceso.

Según la tabla 4.1 son necesarias dos señales de control para el convertidor TL500C : control A y control B que determinarán que parte de la conversión o fase de autocero se lleva a cabo .

MODO	ENTRADA ANALÓGICA	COMPARADOR	CONTROLES A Y B
AUTO ZERO	V	COMPARACION	L L
INTEGRACION DE LA ENTRADA	POSITIVA	-	H -
INTEGRACION DE LA ENTRADA	NEGATIVA	-	L H
		-	-

TABLA 4.1

Pero es necesario detectar los niveles de salida del integrador para establecer si se convierte una señal positiva o negativa .

#### Primera parte .

#### ESTABLECIENDO FUNCIONES DEL HARDWARE

ENTRADAS AL TL500

SALIDAS DEL TL500

CONTROLES A Y B

SEÑAL DEL COMPARADOR

Las señales anteriores se conectan al puerto C del PPI de la siguiente forma:

Al puerto C<sub>L</sub> (02h) la salida del comparador.

El puerto C<sub>H</sub> (02h) es el nibble de comando para los controles A y B.

Segunda parte.

FASE DE AUTOCERO.

Una de las características que ofrece el convertidor utilizado es la posibilidad de inicializar todos los valores de carga de los condensadores que utiliza para las distintas funciones que realiza. Durante esta fase el convertidor trabaja durante 10,000 conteos que realiza el procesador digital, el control de las entradas y salidas del convertidor A/D TL500C lo hará el sistema basado en el  $\mu$ P 8088.

El tiempo aproximado de esta fase esta determinado, por 10,000 conteos a una frecuencia de 120 KHz , aproximandose a los estadares de funcionamiento del TL500C como sigue :

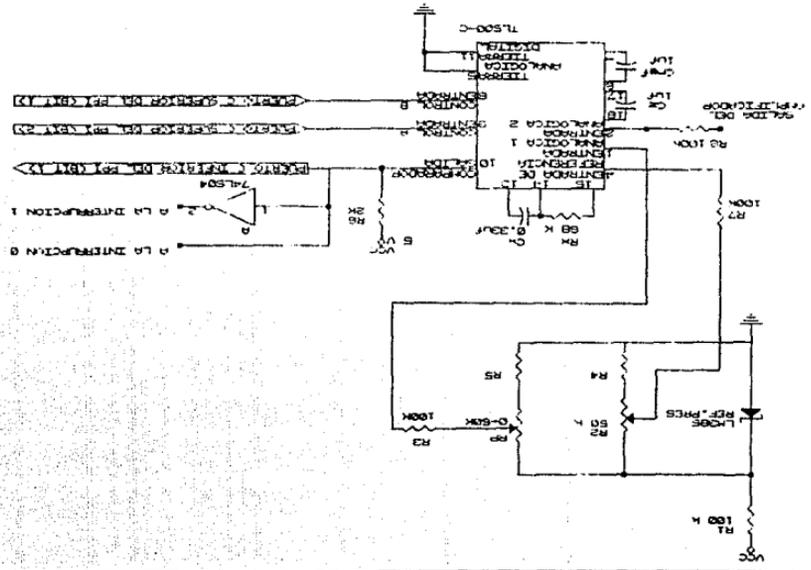
$$T = \frac{1}{120 \text{ KHz}} = 8.33 \times 10^{-6}$$

$$T_{\text{FASE DE AUTOCERO}} = T \times 10,000$$

$$T_{\text{FASE DE AUTOCERO}} = 83.33\text{mS.}$$

El cual puede ser simulado con un contador dentro del 8088 tomando en cuenta los ciclos de reloj necesarios para completar una etapa de conteo de esta magnitud en ensamblador 8088.

FIGURA 4.8 PROCESADOR PARALELO TL500



Durante esta fase el voltaje de referencia es almacenado en CREF, el voltaje de offset del comparador en el condensador del integrador Cx y la suma de los voltajes de offset del buffer del integrador en el capacitor cero Cz. Durante esta fase la salida del comparador se caracteriza por una oscilación de frecuencia y magnitud indeterminada. Esta fase debe comenzar con los controles A y B puestos a cero.

Tercera parte.

FASE DE INTEGRACION DE LA ENTRADA.

Al final de la fase anterior, los controles A y B son puestos a 1 lógico para indicar que ha empezado la fase de integración de la entrada durante un tiempo fijo de 10.000 conteos a una razón marcada por el mismo voltaje. la línea de carga tendrá una pendiente más inclinada cuanto mas sea el voltaje de entrada.

$T_{FASE DE INTEGRACION} = 83.33 \text{ mS.}$

Durante esta fase el nivel lógico del comparador debe rastrearse para saber si es una entrada positiva o negativa y otorgarle el signo correspondiente.

Cuarta parte.

FASE DE INTEGRACION DE LA REFERENCIA.

Al finalizar la etapa de integración de la entrada debe comenzar la fase de integración de la referencia.

Si el comparador esta a 0 deben aplicarse niveles de A=0 y B=1 para aplicar el voltaje de referencia almacenado en CREF en la entrada del buffer. Si la entrada del comparador es 1 corresponde a una entrada negativa y los controles A y B deben ponerse A=1 y B=0. Aplicando así un voltaje negativo al buffer .

Con esta operación se garantiza que el integrador desciende o se eleva hacia el nivel cero apropiadamente. El tiempo requerido para llegar a cero es medido , en este caso por el  $\mu P8088$  , ya que se espera que la salida del integrador cruce cero o que se llegue a un tiempo igual a 20,000 ciclos a una frecuencia de 120 KHz con lo cual se detecta un sobre rango activando la rutina correspondiente .

Debe destacarse que el descenso o ascenso hacia cero es regulado en base al valor del voltaje de referencia. Siendo así una recta de pendiente constante .

$$T_{INT. REF MAX.} = T \times 20.000$$

$$T_{INT. REF} = 166.67 \mu S$$

Los problemas que ahora se plantean son el uso de las rutinas adecuadas primero de conteo y despues de detección de la salida del comparador .

#### 4.6. FUENTE DE ALIMENTACION

En un sistema eléctrico y por ende en un electrónico una parte fundamental es aquella que da la energía necesaria para realizar un trabajo por sencillo que sea. Existen varias formas de proveer una fuente de alimentación y sus características fundamentales son: los niveles de calidad, tipo de señal y en este caso valores de voltaje y corriente que logren mantener el equipo en operación de manera idónea.

El tipo de equipo que se está utilizando en este proyecto exige tres diferentes voltajes debido a la diversidad de tecnología que fue conjuntada y que se resume como sigue:

**ALIMENTACION A 5V  $\pm 5\%$ .** Para el equipo digital: controladores lógica digital de control, el microprocesador, la pantalla y el teclado.

**ALIMENTACION A 12V  $\pm 5\%$ .** Para el controlador analógico TL500C en su alimentación positiva.

**ALIMENTACION A -12V  $\pm 5\%$ .** Para el controlador analógico TL500C en su alimentación negativa.

Lo que se procede a hacer es determinar las características del equipo de alimentación como son corriente y voltaje en el transformador, el puente rectificador, los reguladores de voltaje y los capacitores. Para esto se hace un inventario por voltaje y consumo de corriente de los G.I.'s utilizados.

NUM	C.I.	VOLTAJE	CORRIENTE
1	R2B4	5V	90mA
2	8088	5V	50mA
3	74LS373	5V	4mA
4	74LS373	5V	4mA
5	74LS373	5V	4mA
6	2716*	5V	10mA
7	6116	5V	275mA
8	74LS04	5V	4mA
9	74LS00	5V	4mA
10	74LS32	5V	4mA
11	74LS08	5V	4mA
12	74LS02	5V	4mA
13	74LS259	5V	4mA
14	8255	5V	90mA
15	2716	5V	10mA
16	74LS138	5V	4mA
	8 Displays LED	5V 140x8	1120mA
17	74LS393	5V	4mA
18	74LS244	5V	4mA
19	74LS259	5V	4mA
20	74LS373	5V	4mA
21	74LS138	5V	4mA
22	8259	5V	85mA
	8 74LS373	5V	32mA
	1 74LS240	5V	4mA
23	TL500C	12V	20mA
		-12V	18mA

NOTA 1

Realizando un redondeo para que no existan condiciones críticas de funcionamiento del transformador .

2.0 Amp

\* Si se necesita una mayor capacidad de memoria se cambiara por una 2732 ,consumo de 10 mA ó una 2764 con consumo de 20mA .

NOTA 1: Tomando la corriente en polarización directa de C.D. con 20mA por segmento.

NOTA 2: Todos los controladors y el  $\mu P$  son C.M.O.S. y por tanto su consumo es muy bajo comparado con los circuitos TTL .

Tomando en cuenta una pantalla a base de LED's el transformador que se recomienda tiene las siguientes características: 120v/18v de C.A. con derivaciones para obtener tres señales de valores netos de 18v de C.A. .

El siguiente elemento a elegir son los rectificadores de onda completa ya que deben cumplir con condiciones de voltaje y corriente idénticas al del transformador.

El que más carga recibe es el rectificador que alimentara el voltaje positivo , ya que más del 90% del equipo requiere de alimentación positiva exclusivamente .

Para alimentación positiva de 5 volts se escoge un puente que pueda alimentar 2 AMP a un mínimo de 200V<sub>ppm</sub> de pico inverso. Se escoge entonces un puente SK3647 ó el BR82D.

Es importante que para que el proyecto siga siendo compacto, sino se tiene necesidad, se utilice partes de poco consumo, por eso para la alimentación negativa sera necesario solo un rectificador de 1 AMP y a 200V<sub>ppm</sub>, como el SK9230 ó un equivalente más comercial y para la alimentación positiva de 12 volts se requiere otro rectificador de 1 AMP y a 200V<sub>ppm</sub>.

Estandarizando el tipo de integrados se escoge dividir en 2 la alimentación positiva y utilizar un regulador positivo que pueda manejar 2 AMP de corriente en operación. Para tener 5V con una corriente de operación de 2 AMP hay que regular el voltaje a 5V con un regulador 7805CK. Se regula a 12V para la alimentación positiva del TL500C con un regulador 7812.

En la alimentación negativa un regulador 7912 es elegido, contando con que el consumo de corriente no pasa de unos cuantos miliamperes (la alimentación negativa del TL500C).

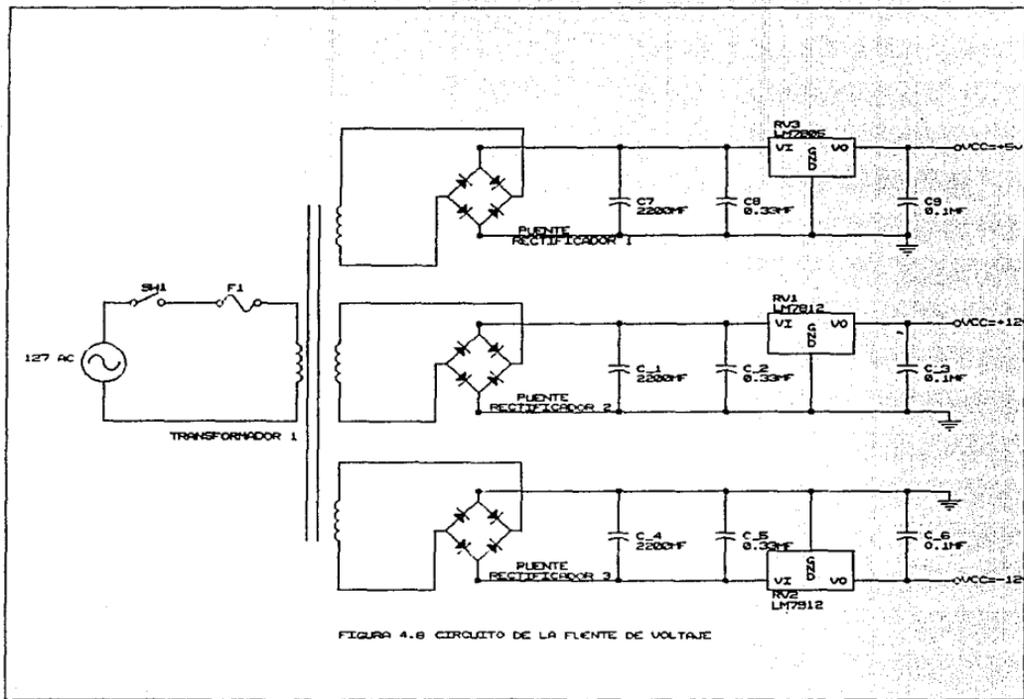


FIGURA 4.8 CIRCUITO DE LA FUENTE DE VOLTAJE

## C A P I T U L O V

### PROGRAMACION

En este capítulo se describen las rutinas que integran la programación de la balanza. Toda la programación esta realizada en lenguaje ensamblador 8088. Una de las razones por la cuales se eligió realizar la programación en lenguaje ensamblador es su mayor rapidez en comparación con algunos lenguajes de alto nivel como el language C o Pascal.

La programación que la balanza necesita esta compuesta de pequeñas rutinas que se vinculan unas con otras para llevar a cabo su correcta operación. Debido a que el programa se basa en el principio de programación estructurada fue posible probar cada rutina individualmente con lo que se facilita su revisión y posibles modificaciones .

La estructura general del programa esta representado en el diagrama de flujo de la figura 5.1 y el programa es mostrado en la figura 5.2.

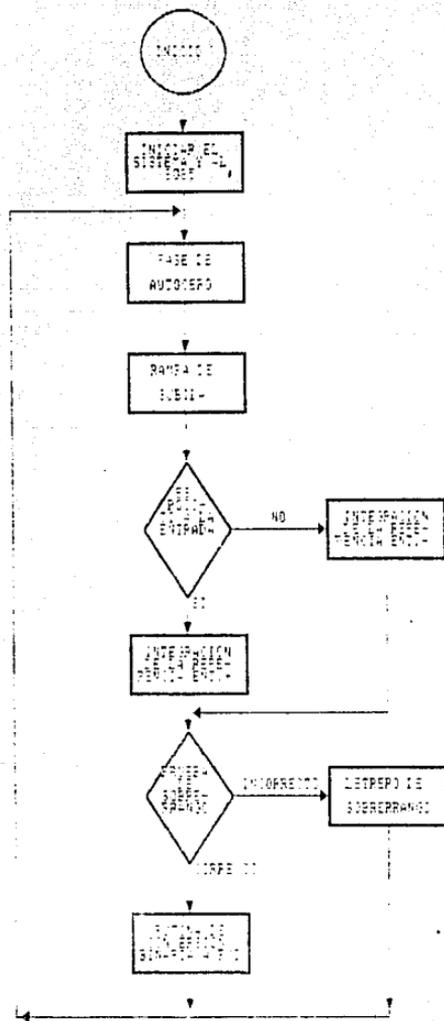


FIGURA 5.1 DIAGRAMA DE FLUJO PARA LA CONVERSION ANALOGICA A DIGITAL

PROGRAMA

FIXED

```

IGUALDADES
SEGMENT AT 0000h
ASSUME DS:FIXED
KBDAT EQU DS:(0088h) ;DATO TECLADO
KBINT EQU DS:(0028h) ;DIRECCION DEL VECTOR
;TECLADO
KINT EQU DS:(0020h) ;DIRECCION DEL VECTOR
;COMPARADOR
KINTN EQU DS:(0024h) ;DIRECCION DEL VECTOR
;COMPARADOR NEGATIVO.
DELAY EQU DS:(0100h) ;DIRECCION DE LA
;CONSTANTE DE RETARDO

TZER EQU 5000
TRAUP EQU 5000
KEYBOARD EQU 00000h ;8255 PORT 0 TECLADO
DISPLED EQU 00001h ;8255 PORT 1 DISPLAY
CONTROL1 EQU 00002h ;8255 PORT 2 CONTROL Y
;COMPARADOR
IOCM1 EQU 00003h ;8255 PORT 3 COMANDO
INTPORT0 EQU 0080h ;8259 No.1 PORT 0
INTPORT1 EQU 0081h ;8259 No.1 PORT 1
MSB EQU DS:(90h)
LSB EQU DS:(91h)
TENTSD EQU DS:(92h)
HNDTHD EQU DS:(93h)
UNTTEN EQU DS:(94h)
DISPLAY1 EQU DS:(00801h)
DISPLAY2 EQU DS:(00811h)
DISPLAY3 EQU DS:(00821h)
DISPLAY4 EQU DS:(00831h)
DISPLAY5 EQU DS:(00841h)
DISPLAY6 EQU DS:(00851h)
FIXED ENDS

```

BOB

START:

```

INICIAR LOS SEGMENTOS
SEGMENT
ASSUME CS:BOB,DS:BOB,SS:BOB
ORG 100h
CLI
MOV AX,0070h ;COLOCA EL INICIO DEL
;STACK EN 0070h Y
MOV SS,AX ;COLOCA EL TOPE
MOV SP,00F0h ;DEL STACK EN 00F0h
MOV AX,0000h ;DICE AL 8088 QUE EL
;SEGMENTO DE DATOS
;ES EN 0000h
MOV DS,AX
INICIALIZAR AL 8255

```

	MOV	AL,91h	:8255a EN MODO 0 CON :PUERTO 0 DE ENTRADA, :PUERTO 1 Y 2 ALTO :DE SALIDA
	OUT	03h,AL	:EL BIT 00010000 DEL :PUERTO 2 CORRESPONDE :A CONTROL B :EL BIT 00100000 DEL :PUERTO 2 :CORRESPONDE A CONTROL A :EL BIT 00000001 DEL :PUERTO 2 CORRESPONDE :A COMPARADOR-IN
INICIO:	MOV	AL,00000000b	:TL500 EN AUTO-ZERO
	OUT	02h,AL	
	MOV	DS:100h),TZER	:TZER DETERMINA EL :TIEMPO QUE :AUTO-CERO OPERA
WAITASEC7:	DEC	WORD PTR DS:100h)	:
	OUT	02h,AL	
	CMP	WORD PTR DS:100h),00h	
	JNZ	WAITASEC7	
RAMPUP:	MOV	AL,00110000b	:INTEGRACION DE LA :ENTRADA
	OUT	02h,AL	:
			:
	MOV	DS:100h),TRAUP	:TRAUP DETERMINA EL :TIEMPO :DE LA RAMPA DE SUBIDA
WAITASEC8:	DEC	WORD PTR DS:100h)	
	OUT	02h,AL	
	CMP	WORD PTR DS:100h),00h	
	JNZ	WAITASEC8	
	IN	AL,02h	:ENTRADA DEL COMPARADOR
	CMP	AL,00h	
	JNZ	POSITIVE	
NEGATIVE:	MOV	AL,00100000b	:INTEGRACION DE LA :REFERENCIA
	OUT	02h,AL	:ENTRADA NEGATIVA
	MOV	AL,17h	:CARGAR ICW1
	OUT	INTPORT0,AL	:EN EL 8259
	MOV	AL,08h	:CARGAR ICW2
	OUT	INTPORT1,AL	:EN EL 8259
	MOV	AL,0Fh	:CARGAR ICW4
	OUT	INTPORT1,AL	:EN EL 8259
	MOV	AL,0Feh	:CARGAR OCW1
	OUT	INTPORT1,AL	:EN EL 8259
	MOV	WORD PTR KINT.OFFSET GETCOMP	
	MOV	FIGURA 5.2(CONTINUACION)	

```

MOV      WORD PTR KINT+2,0FF70h
STI
MOV      AX,0000h
MOV      DS:100hJ,AX
MOV      AL,00000010b
WAITSEC9: INC      WORD PTR DS:100hJ
OUT      02h,AL
CMP      WORD PTR DS:100hJ,03FFFh
JNZ      WAITSEC9
MOV      AL,0Ah
MOV      DISPLAY1,AL
OUT      01h,AL
ADD      AL,20h
MOV      DISPLAY2,AL
OUT      01h,AL
ADD      AL,40h
MOV      DISPLAY3,AL
OUT      01h,AL
ADD      AL,60h
MOV      DISPLAY4,AL
OUT      01h,AL
ADD      AL,80h
MOV      DISPLAY5,AL
OUT      01h,AL
MOV      AL,TENTSD
ADD      AL,0A0h
MOV      DISPLAY6,AL
OUT      01h,AL
JMP      CONTINUA
POSITIVE: MOV      AL,00010000b
OUT      02h,AL
MOV      AL,17h
OUT      INTPORT0,AL
MOV      AL,08h
OUT      INTPORT1,AL
MOV      AL,0Fh
OUT      INTPORT1,AL
MOV      AL,0FDh
OUT      INTPORT1,AL
MOV      WORD PTR KINTN,OFFSET GETCOMP
MOV      WORD PTR KINTN+2,0FF70h
STI
MOV      AX,0000h
MOV      DS:100hJ,AX
MOV      AL,00000001b
WAITSE10: INC      WORD PTR DS:100hJ
OUT      02h,AL
CMP      WORD PTR DS:100hJ,03FFFh
JNZ      WAITSE10
MOV      AL,0Ah
MOV      DISPLAY1,AL
OUT      01h,AL
ADD      AL,20h

```

FIGURA 5.2 <CONTINUACION>

```

MOV     DISPLAY2,AL
OUT     01h,AL
ADD     AL,40h
MOV     DISPLAY3,AL
OUT     01h,AL
ADD     AL,60h
MOV     DISPLAY4,AL
OUT     01h,AL
ADD     AL,80h
MOV     DISPLAY5,AL
OUT     01h,AL
MOV     AL,TENTSD
ADD     AL,0A0h
MOV     DISPLAY6,AL
OUT     01h,AL
JMP     CONTINUA
CONTINUA:
;
;
ORG     700h
GETCOMP:
CONBIT:   NOP
MOV     AX,(0100h)           ;CONVERSION BIT-BCD
MOV     MSB,AH
MOV     LSB,AL
MOV     DX,0010h
MOV     AL,00h
MOV     UNTTEN,AL
MOV     TENTSD,AL
MOV     HNDTHD,AL
MOV     BL,00h
BEGIN:    MOV     AL,UNTTEN
MOV     CL,01h
MOV     BL,AL
AND     AX,000Fh
SUB     AL,05h
JS      AT
ADD     BL,03h
AT:       MOV     AL,BL
AND     AX,00F0h
SUB     AL,50h
JS      BT
BT:       ADD     BL,30h
MOV     UNTTEN,BL
MOV     AL,HNDTHD
MOV     BL,AL
AND     AL,0Fh
SUB     AL,05h
JS      CT
CT:       ADD     BL,03h
MOV     AL,BL
AND     AX,00F0h
SUB     AL,50h
JS      BT
ADD     BL,30h

```

FIGURA 5.2 (CONTINUACION)

GT:	MOV	HNDTHD,BL
	MOV	AL,TENTSD
	MOV	BL,AL
	SUB	AL,05h
	JS	ET
	ADD	BL,03h
ET:	MOV	TENTSD,BL
	PUSH	AX
	MOV	AL,LSB
	SAL	AL,CL
	MOV	LSB,AL
	MOV	AL,MSB
	RCL	AL,CL
	MOV	MSB,AL
	MOV	AL,UNTTEN
	RCL	AL,CL
	MOV	UNTTEN,AL
	MOV	AL,HNDTHD
	RCL	AL,CL
	MOV	HNDTHD,AL
	MOV	AL,TENTSD
	RCL	AL,CL
	MOV	TENTSD,AL
	POP	AX
	DEC	DX
	JNZ	BEGIN
	MOV	CL,04h
	MOV	AL,UNTTEN
	SHL	AL,CL
	ROR	AL,CL
	MOV	DISPLAY1,AL
	OUT	01h,AL
	MOV	AL,UNTTEN
	SHR	AL,CL
	ADD	AL,20h
	MOV	DISPLAY2,AL
	OUT	01h,AL
	MOV	AL,HNDTHD
	SHL	AL,CL
	ROR	AL,CL
	ADD	AL,40h
	MOV	DISPLAY3,AL
	OUT	01h,AL
	MOV	AL,HNDTHD
	SHR	AL,CL
	ADD	AL,60h
	MOV	DISPLAY4,AL
	OUT	01h,AL
	MOV	AL,TENTSD
	SHL	AL,CL
	ROR	AL,CL
	ADD	AL,80h
	MOV	DISPLAY5,AL

FIGURA 5.2 <CONTINUACION>

```
OUT 01h,AL
MOV AL,TENTSD
SHR AL,CL
ADD AL,0A0h
MOV DISPLAY6,AL
OUT 01h,AL
JMP START
IRET
```

THE BOOT CODE

```
ORG 0BF0h
JMP START
DB 00,00,00,00,00,00
DB 00,00,00,00,00,00
ENDS
END START
```

BOB

El programa empieza con la definición de los valores de las constantes y las direcciones de algunas localidades de memoria.

A partir de la etiqueta **START** se deshabilitan las interrupciones y se determinan las localidades de inicio y tope del stack. A continuación se inicializa al PPI 9255 con la palabra de control para operar de la siguiente forma:

**PALABRA DE CONTROL 91.**

A \_\_\_\_\_ Entrada 8 bits \_\_\_\_\_ P<sub>A7</sub>-P<sub>A0</sub>

C                      Salida 4 bits \_\_\_\_\_ P<sub>C7</sub>-P<sub>C4</sub>  
                     Entrada 4 bits \_\_\_\_\_ P<sub>C3</sub>-P<sub>C0</sub>

B \_\_\_\_\_ Salida 8 bits \_\_\_\_\_ P<sub>B7</sub>-P<sub>B0</sub>

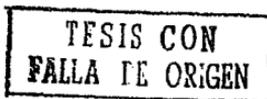
BIT 0 : 1	Puerto	C <sub>L</sub> entrada	} Grupo B.
BIT 1 : 0	Puerto	B salida	
BIT 2 : 0	Modo de Sel.	MODO 0 BASICO	} Grupo A.
BIT 3 : 0	Puerto	C <sub>H</sub> salida	
BIT 4 : 1	Puerto	A entrada	} Grupo A.
BIT 5 : 0	Modo de Sel.	MODO 0 BASICO	
BIT 6 : 0			
BIT 7 : 1	Bandera indicadora de actividad.		

(A 1 lógico hay actividad).

El PPI divide entonces sus funciones así:

Puerto A : 00h teclado.

Puerto B : 01h display.



Puerto C<sub>L</sub>: 02h rastreo . salida del integrador.  
Puerto C<sub>H</sub>: 02h nibble de comando para los  
controles A y B.

En la etiqueta INICIO comienza la fase de autocero, en la cual se aplican niveles altos a los controles A y B. La constante TZER determina la duración de esta fase . el valor de TZER es colocado en una localidad de memoria y esta es decrementada dentro de un bucle , se realiza una comparación con cero para determinar el momento en que el bucle y esta fase deben finalizar.

A partir de la etiqueta RAMPUP inicia la fase de integración de la entrada, la constante TRAUP determina la duración de la rampa de subida, el valor de TRAUP es puesto en una localidad de memoria, esta también es decrementada dentro de un bucle y se realiza una comparación con cero para determinar el fin de la fase. Además se chequea el valor del comparador para definir si el voltaje de entrada es positivo o negativo, en el caso de ser negativo el programa se ramifica a la etiqueta NEGATIVO para aplicar los niveles adecuados a los controles A y B (bajo y alto a las entradas A y B del TL500 respectivamente) para después seguir en la etiqueta CONTINUA; si el voltaje de entrada es positivo el programa va a la etiqueta POSITIVO y se aplican niveles alto y bajo a los controles A y B respectivamente.

La fase de integración de la referencia se realiza a partir de las etiquetas NEGATIVO ó POSITIVO dependiendo de la

polaridad del voltaje de entrada, en esta parte se inicializa al 0259, se habilitan las interrupciones y se empieza un conteo ascendente dentro de un bucle, el valor del conteo es almacenado en una localidad de memoria. Esta fase termina cuando el comparador cambia de estado y es generada una interrupción con lo cual se ejecuta la rutina de conversión de código binario a BCD y su posterior traducción a código de siete segmentos con salida a display. La fase también termina si el conteo excede de 20.000 y entonces es activada una indicación de sobrerango.

#### Conversión Binaria a BCD.

Una técnica estándar de conversión binaria a BCD es aquella del algoritmo de SUMA 3. En las figuras 5.3 y 5.4 se muestra un diagrama de flujo y un ejemplo de este algoritmo. La técnica requiere de un registro que contenga el número de N bits binarios y suficientes registros de 4 bits BCD para contener un número máximo equivalente BCD para el número binario inicial.

La conversión comienza chequeando cada registro para un valor de 5 o mayor, si esta condición existe en uno o todos estos registros, (inicialmente esta condición no puede existir) entonces un 3 es sumado a esos registros donde existe esta condición. El siguiente registro es desplazado a la izquierda con el carry apagado del registro previo, siendo el carry encendido en el siguiente registro. Nuevamente cada registro BCD es chequeado para valores de 5 o mayores. Esta secuencia continúa hasta que los registros han sido desplazados N veces, donde N es el número de bits de la palabra binaria inicial. Los

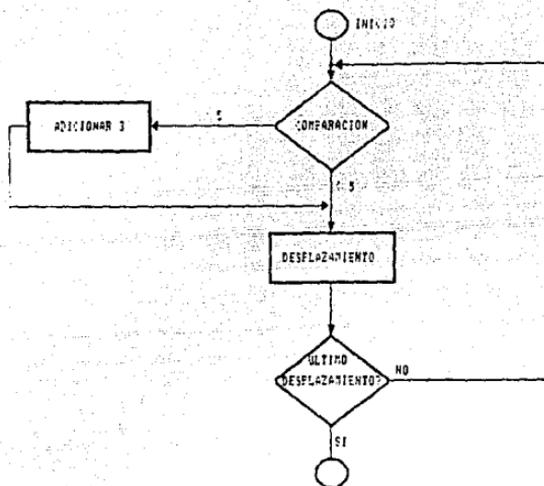


FIGURA 5.3 DIAGRAMA DE FLUJO DE LA CONVERSION BINARIA A BCD.

CENTENAS	DECENAS	UNIDADES	8 BITS BINARIOS	
			1 1 1 1 1 1 1 1	
		1	1 1 1 1 1 1 1	DESPLAZAMIENTO
		1 1	1 1 1 1 1 1	DESPLAZAMIENTO
		1 1 1	1 1 1 1 1	DESPLAZAMIENTO
		1 0 1 0	1 1 1 1 1	ADICIONAR 3
	1	0 1 0 1	1 1 1 1 1	DESPLAZAMIENTO
	1	1 0 0 0	1 1 1 1 1	ADICIONAR 3
	1 1	0 0 0 1	1 1 1 1	DESPLAZAMIENTO
	1 1 0	0 0 1 1	1 1 1	DESPLAZAMIENTO
	1 0 0 1	0 0 1 1	1 1	ADICIONAR 2
1	0 0 1 0	3 1 1 1	1	DESPLAZAMIENTO
1	0 0 1 0	1 0 1 0	1	ADICIONAR 3
1 0	0 1 0 1	0 1 0 1		DESPLAZAMIENTO
2	5	5		TOTAL DE DESPLAZAMIENTOS 8

FIGURA 5.4 EJEMPLO DE CONVERSION BINARIA A BCD

registros BCD contienen el resultante BCD equivalente de la palabra inicial binaria. El ejemplo de la figura 4.4 comienza con una palabra binaria de 8 bits, la cual es convertida al equivalente BCD de 255 por esta técnica. Después de ocho desplazamientos el último bit binario a sido desplazado fuera del registro binario y de los registros de centenas, decenas y unidades que contienen un 255.

En la técnica de conversión de BCD a siete segmentos cada uno de los dígitos completos BCD es convertido a un código de siete segmentos al separar la palabra BCD inferior de la superior y usando el código BCD como la dirección para la tabla de referencia del generador de caracteres. En el caso del dígito BCD superior de cada código BCD, la memoria debe ser desplazada a la izquierda 4 veces para un correcto direccionamiento de la tabla de referencia y finalmente el código de siete segmentos debe ser sacado al puerto de salida.

## CONCLUSIONES DE TESIS.

De la presente exposición se ha podido establecer que para conjuntar un equipo basado en microprocesador que regule, cense ó realice las funciones de una balanza, es preciso reunir técnicas, instrumentos, materiales y además combinarlos adecuadamente para que puedan detectar el desarrollo del fenómeno físico, como en este caso la deflexión de la viga en cantilever como consecuencia de un peso aplicado.

La forma en que se realiza el censado de un evento o un proceso influye invariablemente en la forma en que deba ser tratado para que los datos obtenidos sean útiles. Es así como se requirió una señal analógica que tuviera calidad, es decir, que la señal no llevara "ruido" ó señales parásitas (que es lo que se depuró con la presencia de un filtro pasa bajas) para un convertidor analógico-digital que recibe su señal para conversión de un amplificador de instrumentación cuya ventaja es poder amplificar dentro de rangos variados además de tener la cualidad de separar impedancias tanto del amplificador como del mismo transductor, hacer una preamplificación en modalidad diferencial para después pasar a una amplificación final. Además el amplificador de instrumentación en la configuración utilizada aporta una relación de rechazo al ruido, con lo que queda una señal más pura y precisa.

La calidad de la señal no solo se logra mediante instrumentos electrónicos sino también con el tipo de sensor que se utiliza. El puente de galgas utilizado reúne características de alta sensibilidad, linealidad y en la forma en que se conectó que la señal corresponda a dos efectos sumados, uno de tensión que se presenta en la cara superior de la viga cantilever y otro de compresión de la parte inferior de la misma.

Pueden considerarse al transductor, amplificador y al circuito del procesador analógico como un sistema periférico de un equipo de adquisición de datos, que aunque es la forma más básica de estos sistemas, es un dispositivo basado en microprocesador con las partes esenciales para ubicarlo dentro del universo de los controladores ó incluso de las nanocomputadoras.

La forma en la cual se realizó la interfaz del convertidor analógico-digital, en este caso un procesador analógico TL500C demuestra que no solo es importante la construcción de los equipos, compatibilidad de tecnología ó si fueron hechos para constituir un bloque operativo ex-profeso a una tarea, sino que deben tomarse en cuenta formas ó técnicas de programación, adaptación de señales, como son los controles A y B del TL500C acoplados a 2 bits de un puerto del 8255 y adaptación de las rutinas que forman el cuerpo del programa

principal , ya que deben simular una velocidad de funcionamiento poco más de 10 veces más lenta en el convertidor , comparada con el  $\mu P$  8088.

Al crear la rutinas de conversión analógico-digital aprovechando la forma de funcionamiento del TL500C, el 8255, el 8259 y por supuesto del 8088 además de acoplar la pantalla . el teclado y la memoria se realizaron ejercicios básicos para construir un sistema con base en un  $\mu P$ .

El proyecto , es en sí , la parte esencial de una balanza electronica , al conjuntar un controlador , un convertidor A/D y un transductor de presión .A partir de las bases sentadas pueden darse algunas modificaciones para un desarrollo posterior al ya obtenido.

Lo primero es convertir al controlador en una máquina más poderosa (más memoria o mayor capacidad de interrupciones) siendo entonces un controlador con un periférico cuya función es actuar como balanza electrónica. Para llevar a cabo esta idea es necesario cubrir un plan de necesidades que se deseen que el controlador satisfaga. Se puede incluso integrar una interface de comunicación en formato RS-232C ó en algún otro.

Un punto aparte son las rutinas de atención a pantalla teclado o interrupciones , las cuales pueden depurarse para que realicen acciones más complejas , poderosas o de efectos más variados que los que generan o dan servicio. El controlador se

pueden dedicar a un funcionamiento más especializado tendiente a ofrecer más de las características de las balanzas de precisión de los laboratorios, operaciones estadísticas , pesado de animales vivos , autodiagnóstico' de fallas en el sistema y muchas otras opciones se pueden ofrecer como funciones en un controlador dedicado a operar como balanza .



2) Por su forma:

- Uniaxiales: un solo eje de medida

- Longitudes útiles: 0,3 m/m - 70 m/m.



- Aplicación: Deformaciones sobre un solo eje  
Determinación de fatigas cuando se conocen las direcciones de esfuerzo (una dirección)  
Realización de captadores

- Biaxiales: dos ejes situados, generalmente, a 90°

- Longitudes útiles: 1 - 10 m/m.



- Aplicaciones: Estado de tensiones cuando se conocen las direcciones principales.  
Medida de PAR DE TORSION  
Captadores compensados

- Triaxiales: tres ejes.



- Longitudes útiles: 2 - 15 m/m.

- Aplicaciones: Determinación del estado de tensiones superficiales.

- Multiaxiales: más de tres ejes (normalmente de 4 ejes).



- Aplicaciones: Comprobación de la medición.

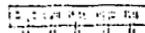
Ejemplo: Con 0-90-45° tendremos una medida,  
con 0-90-135° comprobamos esa medida.  
Un eje de repuesto



- Aplicación: Medidas de tensiones residuales. Es destructiva.

- CALCAS ESPECIALES:

- De varios elementos.



Longitud total 5 m/m.

Aplicación: Determinación de gradientes de tensiones superficiales.

- Espirales;

Dimensiones:  $\varnothing$  15 - 20 mm.



TESIS CON  
FALLA DE ORIGEN

Aplicación: Medida de presiones (diaframas)

- Metálicas (Hilo).

Longitud: 10 m/m.

Fijación por soldadura

Aplicación: Medidas de deformaciones hasta en + 1000°C de tem  
peraturas.

- En base de Asbestos.

Longitud: 10 m/m.

Aplicación hasta + 400°C.

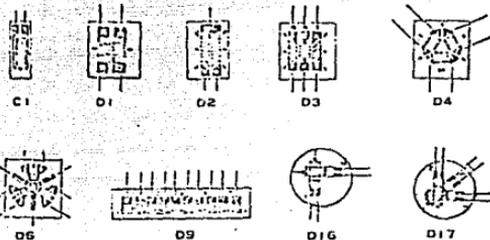
- De gran elongación:

Longitud: 20 m/m.

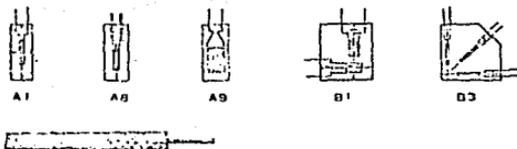
Aplicación: Medidas en la zona plástica ( $\pm 15\%$ ).

List of Gage Patterns

galgas de FILM

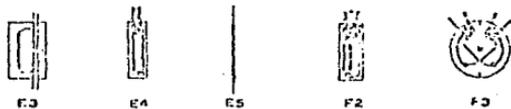


galgas de HILO



H2

galgas SEMICONDUCTORAS



## 8255A/8255A-5 PROGRAMMABLE PERIPHERAL INTERFACE

- MCS-85™ Compatible 8255A-5
- 24 Programmable I/O Pins
- Completely TTL Compatible
- Fully Compatible with Intel® Microprocessor Families
- Improved Timing Characteristics
- Direct Bit Set/Reset Capability Easing Control Application Interface
- 40-Pin Dual In-Line Package
- Reduces System Package Count
- Improved DC Driving Capability

The Intel® 8255A is a general purpose programmable I/O device designed for use with Intel® microprocessors. It has 24 I/O pins which may be individually programmed in 2 groups of 12 and used in 3 major modes of operation. In the first mode (MODE 0), each group of 12 I/O pins may be programmed in sets of 4 to be input or output. In MODE 1, the second mode, each group may be programmed to have 8 lines of input or output. Of the remaining 4 pins, 3 are used for handshaking and interrupt control signals. The third mode of operation (MODE 2) is a bidirectional bus mode which uses 8 lines for a bidirectional bus, and 5 lines, borrowing one from the other group, for handshaking.

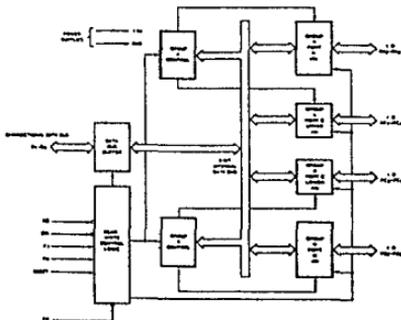
PIN CONFIGURATION



PIN NAMES

PIN	DATA BUS	DIRECTION
A0-A15	DATA BUS	BIDIRECTIONAL
RESET	RESET INPUT	
CS	Chip SELECT	
RD	READ INPUT	
WR	WRITE INPUT	
AD0-A1	PORT ADDRESS	
DATA0	PORT 0 DATA	
DATA1	PORT 1 DATA	
DATA2	PORT 2 DATA	
DATA3	PORT 3 DATA	
VCC	+5 VOLTS	
GND	0 VOLTS	

8255A BLOCK DIAGRAM



## 8255A FUNCTIONAL DESCRIPTION

## General

The 8255A is a programmable peripheral interface (PPI) device designed for use in Intel® microcomputer systems. Its function is that of a general purpose I/O component to interface peripheral equipment to the microcomputer system bus. The functional configuration of the 8255A is programmed by the system software so that normally no external logic is necessary to interface peripheral devices or structures.

## Data Bus Buffer

This 3-state bidirectional 8-bit buffer is used to interface the 8255A to the system data bus. Data is transmitted or received by the buffer upon execution of input or output instructions by the CPU. Control words and status information are also transferred through the data bus buffer.

## Read/Write and Control Logic

The function of this block is to manage all of the internal and external transfers of both Data and Control or Status words. It accepts inputs from the CPU Address and Control buses and in turn, issues commands to both of the Control Groups.

## (CS)

Chip Select. A "low" on this input pin enables the communication between the 8255A and the CPU.

## (RD)

Read. A "low" on this input pin enables the 8255A to send the data or status information to the CPU on the data bus. In essence, it allows the CPU to "read from" the 8255A.

## (WR)

Write. A "low" on this input pin enables the CPU to write data or control words into the 8255A.

(A<sub>0</sub> and A<sub>1</sub>)

Port Select 0 and Port Select 1. These input signals, in conjunction with the RD and WR inputs, control the selection of one of the three ports or the control word registers. They are normally connected to the least significant bits of the address bus (A<sub>0</sub> and A<sub>1</sub>).

## 8255A BASIC OPERATION

A <sub>1</sub>	A <sub>0</sub>	RD	WR	CS	INPUT OPERATION (READ)
0	0	0	1	0	PORT A = DATA BUS
0	1	0	1	0	PORT B = DATA BUS
1	0	0	1	0	PORT C = DATA BUS
					OUTPUT OPERATION (WRITE)
0	0	1	0	0	DATA BUS = PORT A
0	1	1	0	0	DATA BUS = PORT B
1	0	1	0	0	DATA BUS = PORT C
1	1	1	0	0	DATA BUS = CONTROL
					DISABLE FUNCTION
X	X	X	X	1	DATA BUS = 3-STATE
1	1	0	1	0	ILLEGAL CONDITION
X	X	1	1	0	DATA BUS = 3-STATE

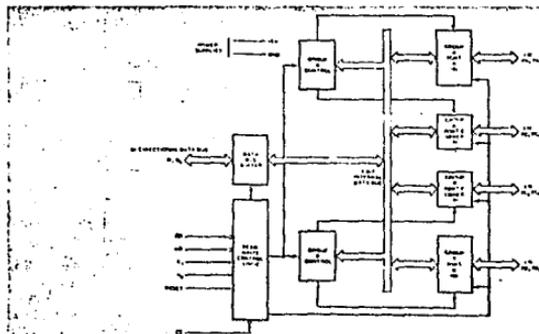


Figure 1. 8255A Block Diagram Showing Data Bus Buffer and Read/Write Control Logic Functions

## 8088 8-BIT HMOS MICROPROCESSOR

- 8-Bit Data Bus Interface
- 16-Bit Internal Architecture
- Direct Addressing Capability to 1 Mbyte of Memory
- Direct Software Compatibility with 8085
- 14-Word by 16-Bit Register Set with Symmetrical Operations
- 24 Operand Addressing Modes
- Byte, Word, and Block Operations
- 8-Bit and 16-Bit Signed and Unsigned Arithmetic In Binary or Decimal, including Multiply and Divide
- Compatible with 8155-2, 8755A-2 and 8185-2 Multiplexed Peripherals

The Intel®8088 is a new generation, high performance microprocessor implemented in N-channel, depletion load, silicon gate technology (HMOS), and packaged in a 40-pin CerDIP package. The processor has attributes of both 8 and 16-bit microprocessors. It is directly compatible with 8086 software and 8080/8085 hardware and peripherals.

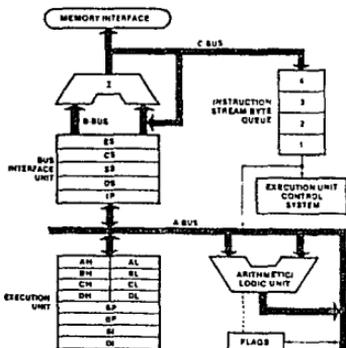


Figure 1. 8088 CPU Functional Block Diagram



Figure 2. 8088 Pin Diagram

## ABSOLUTE MAXIMUM RATINGS\*

Ambient Temperature Under Bias ..... 0°C to 70°C  
Storage Temperature ..... - 65°C to + 150°C  
Voltage on Any Pin with  
Respect to Ground ..... - 1.0 to + 7V  
Power Dissipation ..... 2.5 Watt

\*NOTICE: Stresses above those listed under "Absolute Maximum Ratings" may cause permanent damage to the device. This is a stress rating only and functional operation of the device at these or any other conditions above those indicated in the operational sections of this specification is not implied. Exposure to absolute maximum rating conditions for extended periods may affect device reliability.

## D.C. CHARACTERISTICS

Notes:  $T_A = 0^\circ\text{C}$  to  $70^\circ\text{C}$ ,  $V_{CC} = 5V \pm 10\%$

Symbol	Parameter	Min.	Max.	Units	Test Conditions
$V_{IL}$	Input Low Voltage	-0.5	+0.8	V	
$V_{IH}$	Input High Voltage	2.0	$V_{CC} + 0.5$	V	
$V_{OL}$	Output Low Voltage		0.45	V	$I_{OL} = 2.0\text{ mA}$
$V_{OH}$	Output High Voltage	2.4		V	$I_{OH} = 400\ \mu\text{A}$
$I_{CC}$	Power Supply Current		340	mA	
$I_{LI}$	Input Leakage Current		$\pm 10$	$\mu\text{A}$	$0V < V_{IN} < V_{CC}$
$I_{LO}$	Output Leakage Current		$\pm 10$	$\mu\text{A}$	$0.45V < V_{OUT} < V_{CC}$
$V_{CL}$	Clock Input Low Voltage	-0.5	+0.6	V	
$V_{CH}$	Clock Input High Voltage	3.9	$V_{CC} + 1.0$	V	
$C_{IN}$	Capacitance of Input Buffer (All input except $AD_3$ - $AD_7$ , $RO$ / $GT$ )		15	pF	$f_c = 1\text{ MHz}$
$C_{IO}$	Capacitance of I/O Buffer ( $AD_0$ - $AD_2$ , $RO$ / $GT$ )		15	pF	$f_c = 1\text{ MHz}$







PRELIMINARY

## 8259A/8259A-2/8259A-8 PROGRAMMABLE INTERRUPT CONTROLLER

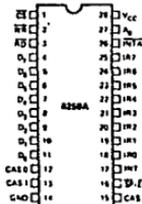
- 8086/8088 Compatible
- MCS-80/85™ Compatible
- Eight-Level Priority Controller
- Expandable to 64 Levels
- Programmable Interrupt Modes
- Individual Request Mask Capability
- Single +5V Supply (No Clocks)
- 28-Pin Dual-In-Line Package

The Intel® 8259A Programmable Interrupt Controller handles up to eight vectored priority interrupts for the CPU. It is cascadeable for up to 64 vectored priority interrupts without additional circuitry. It is packaged in a 28-pin DIP, uses NMOS technology and requires a single +5V supply. Circuitry is static, requiring no clock input.

The 8259A is designed to minimize the software and real time overhead in handling multi-level priority interrupts. It has several modes, permitting optimization for a variety of system requirements.

The 8259A is fully upward compatible with the Intel® 8259. Software originally written for the 8259 will operate the 8259A in all 8259 equivalent modes (MCS-80/85, Non-Buffered, Edge Triggered).

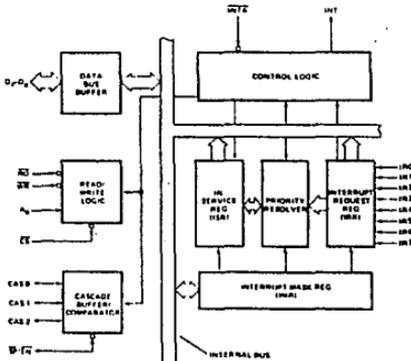
### PIN CONFIGURATION



### PIN NAMES

D <sub>0</sub> , D <sub>1</sub>	DATA BUS DIRECTIONAL
RD	READ INPUT
WR	WRITE INPUT
A <sub>16</sub>	COMMAND SELECT ADDRESS
CS	CHIP SELECT
CAS0, CAS1	CASCADE LINES
DP, EN	SLAVE PROGRAM ENABLE BUFFER
INT	INTERRUPT OUTPUT
INTA	INTERRUPT ACKNOWLEDGE INPUT
IR0-IR7	INTERRUPT REQUEST INPUTS

### BLOCK DIAGRAM





## PIN FUNCTIONS

NAME	I/O	PIN #	FUNCTION
V <sub>CC</sub>	I	28	+5V Supply
GND	I	14	Ground
CS	I	1	Chip Select: A low on this pin enables RD and WR communication between the CPU and the 8259A. INTA functions are independent of CS.
WR	O	2	Write: A low on this pin when CS is low enables the 8259A to accept command words from the CPU.
RD	I	3	Read: A low on this pin when CS is low enables the 8259A to release status onto the data bus for the CPU.
D <sub>7</sub> -D <sub>0</sub>	I/O	4-11	Bidirectional Data Bus: Control, status and interrupt-vector information is transferred via this bus.
CAS <sub>0</sub> -CAS <sub>2</sub>	I/O	12, 13, 15	Cascade Lines: The CAS lines form a private 8259A bus to control a multiple 8259A structure. These pins are outputs for a master 8259A and inputs for a slave 8259A.
SPEN	I/O	16	Slave Program/Enable Buffer: This is a dual function pin. When in the Buffered Mode it can be used as an output to control buffer transceivers (EN). When not in the buffered mode it is used as an input to designate a master (SP = 1) or slave (SP = 0).

NAME	I/O	PIN #	FUNCTION
INT	O	17	Interrupt: This pin goes high whenever a valid interrupt request is asserted. It is used to interrupt the CPU, thus it is connected to the CPU's interrupt pin.
IR <sub>0</sub> -IR <sub>7</sub>	I	18-25	Interrupt Request: Asynchronous inputs. An interrupt request can be executed by raising an IR input (low to high) and holding it high until it is acknowledged (Edge Triggered Mode), or just by a high level on an IR input (Level Triggered Mode).
INTA	I	26	Interrupt Acknowledge: This pin is used to enable 8259A interrupt vector data onto the data bus. This is done by a sequence of interrupt acknowledge pulses issued by the CPU.
A <sub>0</sub>	I	27	A0 Address Line: This pin acts in conjunction with the CS, WR, and RD pins. It is used by the 8259A to decipher between various Command Words the CPU writes and status the CPU wishes to read. It is typically connected to the CPU A0 address line (A1 for 8008/8086).

## ABSOLUTE MAXIMUM RATINGS\*

Ambient Temperature Under Bias	..... -40°C to 85°C
Storage Temperature	..... -65°C to +150°C
Voltage on Any Pin	..... with Respect to Ground ..... -0.5V to +7V
Power Dissipation	..... 1 Watt

## \*COMMENT:

Stresses above those listed under "Absolute Maximum Ratings" may cause permanent damage to the device. This is a stress rating only and functional operation of the device at these or other conditions above those indicated in the operational sections of this specification is not implied.

## D.C. CHARACTERISTICS

T<sub>A</sub> = 0°C to 70°C, V<sub>CC</sub> = 5V ± 10% (8259-A), V<sub>CC</sub> = 5V ± 10% (8259A)

Symbol	Parameter	Min.	Max.	Units	Test Conditions
V <sub>IL</sub>	Input Low Voltage	-0.5		V	
V <sub>IH</sub>	Input High Voltage	2.0	V <sub>CC</sub> + 0.5V	V	
V <sub>OL</sub>	Output Low Voltage		0.45	V	I <sub>OL</sub> = 2.2 mA
V <sub>OH</sub>	Output High Voltage	2.4		V	I <sub>OH</sub> = -400 μA
V <sub>OMPH</sub>	Interrupt Output High Voltage	3.5		V	I <sub>OH</sub> = -100 μA
	Voltage	2.4		V	I <sub>OH</sub> = -400 μA
I <sub>IL</sub>	Input Load Current		10	μA	V <sub>IN</sub> = V <sub>CC</sub> to 0V
I <sub>OL</sub>	Output Leakage Current		-10	μA	V <sub>OUT</sub> = 0.45V
I <sub>CC</sub>	V <sub>CC</sub> Supply Current		85	mA	
I <sub>IR</sub>	IR Input Load Current		-300	μA	V <sub>IN</sub> = 0
			10	μA	V <sub>IN</sub> = V <sub>CC</sub>



## 2716 16K (2K x 8) UV ERASABLE PROM

- **Fast Access Time**
  - 350 ns Max. 2716-1
  - 390 ns Max. 2716-2
  - 450 ns Max. 2716
  - 490 ns Max. 2716-5
  - 650 ns Max. 2716-6
- **Single +5V Power Supply**
- **Low Power Dissipation**
  - 525 mW Max. Active Power
  - 132 mW Max. Standby Power
- **Pin Compatible to Intel® 2732 EPROM**
- **Simple Programming Requirements**
  - Single Location Programming
  - Programs with One 50 ms Pulse
- **Inputs and Outputs TTL Compatible during Read and Program**
- **Completely Static**

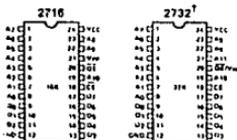


The Intel® 2716 is a 16,384-bit ultraviolet erasable and electrically programmable read-only memory (EPROM). The 2716 operates from a single 5-volt power supply, has a static standby mode, and features fast single address location programming. It makes designing with EPROMs faster, easier and more economical.

The 2716, with its single 5-volt supply and with an access time up to 350 ns, is ideal for use with the newer high performance +5V microprocessors such as Intel's 8085 and 8086. A selected 2716-5 and 2716-6 is available for slower speed applications. The 2716 is also the first EPROM with a static standby mode which reduces the power dissipation without increasing access time. The maximum active power dissipation is 525 mW while the maximum standby power dissipation is only 132 mW, a 75% savings.

The 2716 has the simplest and fastest method yet devised for programming EPROMs — single pulse TTL level programming. No need for high voltage pulsing because all programming controls are handled by TTL signals. Program any location at any time—either individually, sequentially or at random, with the 2716's single address location programming. Total programming time for all 16,384 bits is only 100 seconds.

### PIN CONFIGURATION



† Refer to 2732 data sheet for specifications

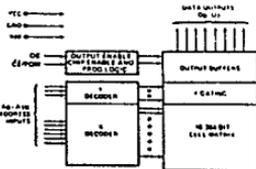
### PIN NAMES

Pin No.	Symbol	ADDRESS
1	VCC	Chip Enable
2	GND	Chip Enable
3	OE	Output Enable
4	CE	Output Enable

### MODE SELECTION

MODE	Pin#	CE Input 100	OE Input 100	V <sub>cc</sub> 12V	V <sub>cc</sub> 12V	OUTPUT 15-17, 19-21, 23-25
Read		V <sub>IL</sub>	V <sub>IL</sub>	-5	+5	Output
Standby		V <sub>OH</sub>	Open	+5	+5	High Z
Program		Pulsed V <sub>IL</sub> 40 V <sub>OH</sub>	V <sub>OH</sub>	+12V	+5	Data
Program Wait		V <sub>IL</sub>	V <sub>IL</sub>	+12V	+5	Output
Program End		V <sub>IL</sub>	V <sub>OH</sub>	+12V	+5	High Z

### BLOCK DIAGRAM



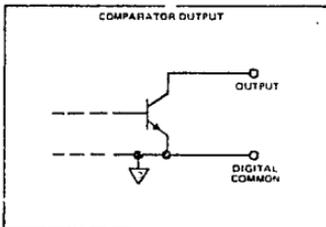
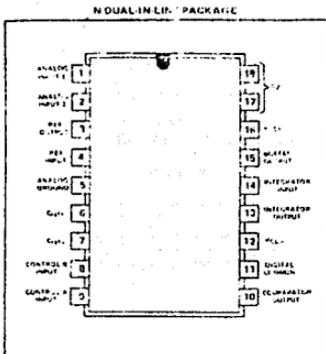
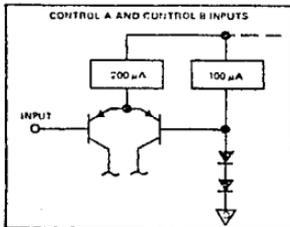


## ANALOG PROCESSORS

### description of analog processors

The TL500C and TL501C analog processors are designed to automatically compensate for internal zero offsets, integrate a differential voltage at the analog inputs, integrate a voltage at the reference input in the opposite direction, and provide an indication of zero-voltage crossing. The external control mechanism may be a microcomputer and software routine, discrete logic, or a TL500C or TL501C controller. The TL500C and TL501C are designed primarily for simple, cost-effective, dual slope analog to digital converters. Both devices feature true differential analog inputs, high input impedances, and an internal reference voltage source. The TL500C provides 4 1/2-digit readout accuracy when used with a precision external reference voltage. The TL501C provides 100ppm linearity error and 3 1/2-digit accuracy capability. These devices are manufactured using TI's advanced technology to produce JFET, MOSFET, and bipolar devices on the same chip. The TL500C and TL501C are intended for operation over the temperature range of 0°C to 70°C.

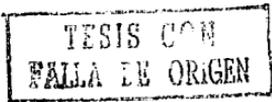
### schematics of inputs and outputs



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

Positive supply voltage, $V_{CC}$ (see Note 1)	+18 V
Negative supply voltage, $V_{CC}$	-18 V
Input voltage, $V_I$	$\pm V_{CC}$
Comparator output voltage range (see Note 2)	0 V to $V_{CC}$
Comparator output sink current (see Note 2)	20 mA
Buffer, reference, or integrator output source current (see Note 2)	10 mA
Operating free-air temperature range	-40°C to 85°C
Storage temperature range	-65°C to 125°C

\*As shown, 1) Input voltage, except differential voltages are with respect to the analog ground and digital common pins tied together.  
2) Buffer, reference, and integrator outputs are not under current protection.





## ANALOG PROCESSORS

### recommended operating conditions

	MIN	NOM	MAX	UNIT
Positive supply voltage, VCC+	7	12	15	V
Negative supply voltage, VCC-	-8	-12	-15	V
Reference input voltage, V <sub>REF</sub> (I)	0.1		5	V
Analog input voltage, V <sub>I</sub>			5	V
Differential analog input voltage, V <sub>ID</sub>			10	V
Peak positive integrator output voltage, V <sub>OM+</sub>			+5	V
Peak negative integrator output voltage, V <sub>OM-</sub>			-5	V
Full scale input voltage			2 V <sub>REF</sub>	V
Autozero and reference capacitors, C <sub>2</sub> and C <sub>REF</sub>	0.2			nF
Temperature capacitor, C <sub>T</sub>	0.2			nF
Integrator resistor, R <sub>I</sub>	15		100	kΩ
Integrator time constant, τ <sub>INTEG</sub>	see Note 4			s
Freezer operating temperature, T <sub>A</sub>	0		75	°C
Maximum conversion rate (see Figure 7)	4 1/2 D		15	CONV/SEC
	1 1/2 C		15.0	CONV/SEC

system electrical characteristics at VCC = ±12 V, T<sub>A</sub> = 25°C (unless otherwise noted) see Figure 3

PARAMETER	TEST CONDITIONS	TL501C			TL500C			UNITS
		MIN	TYP	MAX	MIN	TYP	MAX	
Zero error		50	100		50	100	mV	
Linearity error relative to full scale		0.02	0.05		0.02	0.05	%	
Full scale temperature coefficient	T <sub>A</sub> = 0° to 75°C	1			1		ppm/V	
Temperature coefficient of zero error	T <sub>A</sub> = 0°C to 75°C	1			1		ppm/V/°C	
Rollover error		200	250		150	200	mV	
ET converter peak-to-peak input noise voltage		10			10		mV	
Analog input resistance	pin 1 to 2	100			100		Ω	
Common-mode rejection ratio	V <sub>REF</sub> = 0 V	50			50		dB	
Common-mode rejection ratio	V <sub>REF</sub> = 5 V	50			50		dB	
Current into analog input	V <sub>I</sub> = 15 V	50			50		μA	
Supply voltage rejection ratio		50			50		dB	

Note 4: The minimum integrator time constant may be found by use of the following formula:

$$\tau_{\text{INTEG}} = \frac{V_{ID}(\text{full scale})}{\text{Minimum } R_I C_T} = \frac{V_{ID}(\text{full scale})}{V_{OM} - V_{REF} A_{OL}}$$

where

V<sub>ID</sub> = voltage at pin 1 with respect to pin 2

V<sub>OM</sub>(full) = voltage at pin 2 with respect to analog output

t<sub>I</sub> = input integration time seconds

TESIS CON  
FALLA DE ORIGEN

## BIBLIOGRAFIA

- **Malvino, Albert Paul Ph.D.**  
**Principios de electrónica 3<sup>a</sup> ed. USA**  
Mc.Graw Hill 1986  
México 2<sup>a</sup> ed. 1986
- **Hhatek, Eugene R.**  
**A user's handbook of D/A and D/A converters**  
Wiley-Interscience publication  
N.Y. USA 1976
- **Morgan, Christopher L**  
**Waite, Mitchell**  
**Introducción al µP 8086/8088 (16 bits)**  
Mc Graw Hill México 1988  
Mc Graw Hill USA 1982
- **Grossblatt, Robert**  
**The 8088 project book**  
TAB books Inc. USA 1988
- **Tower, T.D.**  
**Tablas universales Towers. Selección de Transistores**  
Ed Marcombo 2<sup>a</sup> ed. Barcelona 1981
- **Belove, Charles**  
**Schilling, Donald S.**  
**Electronic circuits. Discrete and integrate**  
Ed Marcombo 2<sup>a</sup> ed.  
Mc Graw Hill N.Y. USA 1987

- Varios autores

Serie mundo electronica Transductores y medidores electronicos

Ed. Marcombo Botxareu Barcelona

1983. 2<sup>a</sup> Edición

- National Semiconductor

Memory databook 1988 edition

National Semiconductor USA

2900 Semiconductor drive P.O.Box 58090

Santa Clara CA 95052-8090

- National Semiconductor

Linear 1 databook 1988. Voltage regulators Op. Amp.

Voltage comparators Instrument amplifiers Surface mount.

National Semiconductor USA

2900 Semiconductor drive P.O.Box 58090

Santa Clara CA 95052-8090

- National Semiconductor

Linear 2 databook 1988. Active filters Analog switches/

multiplexers AND D/A Sample and Hold sensors Voltage references Surface mount.

National Semiconductor USA

2900 Semiconductor drive P.O.Box 58090

Santa Clara CA 95052-8090

- RCA Solid State division

Reliable replacement semiconductors.S.K.Series

RCA Solid State division USA 1986

Box 3200 Somerville N.J. 08876



- \* Mano, M. Morris  
 Logica digital y diseño de computadoras  
 USA 1979 Prentice Hall Inc.  
 Mexico 1982 Prentice Hall Hisp. S.A. 1ª Ed.
- \* Williams, Arthur B.  
 Conversion AND y DNA Circuitos logicos SSI, circuitos  
 lógicos MSI.  
 Mc Graw Hill 1984, México
- \* Aldrige, Don  
 Analog to digital conversion techniques with the  
 MC6800  $\mu$ P system.  
 Motorola Inc. USA 1975.
- \* I.E.E. Irot Equipos Electronicos  
 Apuntes de extensometria  
 Seminario sobre "Ensayos y medidas mecanicas"  
 IEE. Irot Equipos Electronicos  
 Bilbao #6 c/Vicente Garamendi #6 Tel :424 7489 424 7477  
 Madrid #28 c/Villafranca.28 Tel : 245 1743.
- \* Ingenieria Española de Procesos y Control S.A.  
 Apuntes de :
  - ENSAYOS MECANICOS
  - INTRODUCCION A LOS SISTEMAS DE INSTRUMENTACION.
  - EXTENSOMETRIA I :GENERALIDADES
  - EXTENSOMETRIA II :CIRCUITOS DE MEDIDA
  - EXTENSOMETRIA III :INTERPRETACION DE MEDIDAS
 EN STRAIN GAGES.  
 Ingenieria Española de Procesos y Control S.A.

Robert G. Sempel

Transducer, Sensors and detectors

Reston Publishing, Inc.

A Prentice Hall Company

Reston Virginia 22090, 1983

TECIS CON  
ALLA LE ORIGIN