

ORIGINAL

PROYECTO

VISUALIZADOR DE AUDIO

ALUMNO: Soledad Pérez Estrada.

TUTOR: Manuel Cubero Enrici.

ORIGINAL

-- PROLOGO --.

Muy a pesar nuestro, la realización -- práctica de este proyecto no ha podido llevarse a cabo debido a la falta de liquidez económica del Departamento de Imagen y Sonido de la Carrera de Telecomunicaciones de la Escuela Universitaria Politécnica.

Nos sentíamos muy ilusionados de poder montar un aparato de esta complejidad y poder presentar no sólo una buena exposición teórica del mismo sino apoyarla y defenderla con un montaje esmerado y de gran calidad.

Hemos esperado esta subvención hasta donde nos ha sido posible, adelantando incluso el diseño de las placas, el programa del ordenador, la localización de diferentes comercios donde solicitar el presupuesto de parte de los componentes, la comunicación con distintos distribuidores de la Península a fin de conseguir información detallada de los componentes de alta precisión que no pudimos adquirir en nuestras islas,...

El tiempo ha pasado y aunque teniendo todo preparado para que una vez llegado el dinero pudiéramos comprar el material y realizar el montaje y ajuste del aparato, estas esperanzas han sido frustradas una vez que se ha abierto el plazo de Matrícula para la presentación de Proyectos y no ha llegado esta ayuda económica. Sin embargo, esta decepción no nos ha hecho desistir de incluir en la exposición teórica el apartado correspondiente al --

diseño y construcción de las placas de nuestro analizador. Asimismo presentaremos las plantillas de las placas, el programa del ordenador,... y la explicación del montaje y del ajuste del mismo.

Esta decisión la hemos tomado pensando en que si no hemos podido montarlo nosotros, quede a disposición de cualquiera que tenga interés en comprobar nuestros datos y observaciones construyendo un aparato que sin duda alguna le reportará una experiencia muy positiva debido a la envergadura del mismo.

Agradecemos a nuestro tutor, D. Manuel Cubero Enrici, el apoyo e interés que mostró en todo momento. Al igual que nosotros, quería que este proyecto se llevara a la práctica y nos consta que hizo repetidas gestiones para que así fuera, sin embargo nadie puede ocultar la mala situación económica que atraviesa nuestra universidad y nunca descartábamos la indeseable posibilidad de que nuestro objetivo no pudiera llevarse a cabo.

## INDICE GENERAL

### PAGINA

Prólogo.....

#### EXPOSICION TEORICA

=====

CAPITULO I: .....

I.1. EL SONIDO.

I.2. PROPAGACION DEL SONIDO.

I.3. EL OIDO HUMANO.

I.4. FACTORES DE LA AUDICION.

I.5. LAS NOTAS MUSICALES.

I.6. LA ESCALA TEMPERADA.

I.7. REGISTROS.

CAPITULO II: .....

II.1. ANALIZADOR DE ESPECTRO AUDIO.

II.2. ANALIZADOR EN TIEMPO REAL.

II.3. FILTROS COMPARTIDOS.

II.4. ANALIZADORES ESTATICOS.

II.5. ANALIZADORES LOGICOS.

CAPITULO III: .....

III.1. ANALIZADOR DE ESPECTRO DE ALTA TEC-  
NOLOGIA: SU FUNCION.

III.2. PRESENTACION Y CONTROLES.

III.3. CARACTERISTICAS NORMALES.

III.4. FORMAS DE REPRESENTACION.

## EXPOSICION TECNICA

=====

MONTAJE: ANALIZADOR ESPECTRO DE AUDIO .....

INTRODUCCION.

1. DIAGRAMA DE BLOQUES.

2. FUENTE DE ALIMENTACION.

2.1. OPERACIONES PRINCIPALES.

2.2. ANALISIS APROXIMADO DEL FILTRO DE CONDENSADOR.

2.3. FUNCION Y JUSTIFICACION DE COMPONENTES.

3. CIRCUITO DE ENTRADA.

4. CIRCUITO ATENUADOR.

5. CIRCUITO DE PROTECCION.

6. CALCULO DE LOS CONDENSADORES DE ACOPLO.

7. LOS FILTROS.

7.1. FILTRO PASABANDA RESONANTE ACTIVO.

7.2. DESCRIPCION TEORICA.

7.3. DEMOSTRACION MATEMATICA.

7.4. DEDUCCION DE LAS FORMULAS PARA EL CALCULO DE LOS COMPONENTES.

7.5. DEDUCCION DE LAS FORMULAS PARA EL CALCULO DE LAS FRECUENCIAS.

7.6. CALCULO DE LOS COMPONENTES DE CADA FILTRO.

8. LOS RECTIFICADORES.

8.1. CALCULO DE LAS RESISTENCIAS DE CARGA.

9. EL VISUALIZADOR DE DIODOS LEDS.

10. GENERADOR DE RUIDO ROSA.

10.1. RUIDO ROSA.

10.2. GENERADOR DE RUIDO.

11. VISUALIZACION POR ORDENADOR.

11.1. INTRODUCCION.

## 11.2. PROGRAMA.

### APENDICE I; TRATAMIENTO DE LOS COMPONENTES.

I.1. COMPONENTES DE DOS TERMINALES.

I.2. CONFORMADO DE TERMINALES.

I.3. ESTAÑADO PREVIO.

I.4. COMPONENTES DE MAYOR POTENCIA.

I.5. MONTAJE DE HILOS Y CABLES.

I.6. TUBOS AISLANTES.

I.7. MONTAJE DE DISIPADORES.

### APENDICE II: DISEÑO DE CIRCUITOS IMPRESOS.

II.1. EL SOPORTE.

II.2. LAS PLACAS.

II.3. DIMENSIONES.

II.4. EL DISEÑO.

II.5. MECANIZADO DE PLACAS.

II.6. LOS TALADROS.

II.7. TERMINACION MECANICA.

II.8. TRANSCRIPCION DEL CIRCUITO.

II.9. LAMINAS Y CINTAS.

II.10. ROTULADORES.

II.11. FOTOGRAFADO.

II.12. ACABADO DEL CIRCUITO.

II.13. TIEMPO DEL PROCESO.

II.14. ACABADO FINAL.

### APENDICE III: CONSTRUCCION DE LAS PLACAS.

...DISEÑO.

...CONSTRUCCION.

...MONTAJE Y AJUSTE.

CARACTERISTICAS DE INTEGRADOS, ESPECIFICACIONES REFERENTES AL ORDENADOR, PLANOS, PLANTILLAS DE LAS PLACAS, DISEÑOS EN PAPEL MILIMETRADO Y PRESUPUESTO.

EXPOSICION TEORICA



## CAPITULO I

### 1.1. EL SONIDO

El sonido es uno de los fenómenos de la Naturaleza que resulta más familiar a los seres humanos. Quizás debido a ello, generalmente pasan desapercibidos la gran variedad de matices y posibilidades que presenta, detalles que en resumen, son muy útiles a la hora de comprender cómo se genera, se transmite y se percibe por el oído.

Normalmente el sonido se caracteriza por tres factores que definen sus propiedades más importantes. Son la Intensidad, el Tono y el Timbre. La Intensidad es el valor máximo de la presión sonora, transmitida por el aire en forma de ondas sinusoidales, que causa directamente la sensación de escucha al incidir sobre las regiones sensibles del oído.

Esta presión sonora se mide sobre la base de cierto nivel tomado como referencia y calculando el cociente entre ambos.

El resultado obtenido expresado en decibelios nos dará el valor buscado en unidades denominadas SPL. El nivel de presión, tomado como referencia, tiene un valor de  $1996 \times 10^{-6} \text{ kg/m}^2$  y se define para una frecuencia de 1 KHz.

Desde el nivel cero de presión en unidades SPL hasta los 120 dB correspondientes a una sensación dolorosa al ser escuchado, existe una amplia gama de niveles sonoros que pueden ser asimilables a diversas situaciones habituales en nuestra

vida, con objeto de que pueda determinarse con una cierta aproximación, qué intensidad sonora existe en cada caso, sin necesidad de ningún instrumento de medida. Así, por ejemplo, un nivel de 20 dB equivale a una presión sonora absoluta de  $2 \times 10^{-5}$  kg/m<sup>2</sup>, puede corresponder al nivel sonoro existente en un jardín situado lejos de cables o carreteras con tráfico y sin ninguna fuente sonora artificial que eleve anormalmente dicho nivel. Puede considerarse, por lo tanto, a esta intensidad sonora como muy débil.

Otro caso de mayor presión sonora es el nivel de 40 dB, que representa aproximadamente  $2 \times 10^{-4}$  Kg/m<sup>2</sup>, es decir, diez veces superior al anterior. Ello equivale al nivel acústico de una vivienda doméstica vacía, únicamente sujeta a los ruidos transmitidos por las paredes, suelos y techos.

Y ya entrando en niveles algo más elevados podemos tomar el de 60 dB, que supone una presión de  $2 \times 10^{-3}$  Kg/m<sup>2</sup>, lo que equivale a la intensidad sonora generada por una conversación entre varias personas que ninguna de ellas eleve el nivel de su voz.

Otros niveles mucho más elevados corresponden a valores del orden de los 100 dB que en términos de presión supondría la de  $0.2$  Kg/m<sup>2</sup>,  $200$ g/m<sup>2</sup>, la sensación acústica de este nivel sería parecida a la del sonido producido en el interior de una fábrica en plena actividad.

Por último, puede indicarse un nivel de alrededor de 120 dB como el máximo que podría escu-

char el oído humano sin que se produzcan en él lesiones graves, aunque percibiendo una sensación dolorosa. Esta intensidad equivalente en términos de presión a  $2 \text{ Kg/m}^2$  puede asimilarse al ruido producido por un avión reactor en el momento del despegue.

Otro de los factores característicos del sonido, como es el tono, está definido por la frecuencia en hercios de la componente fundamental de éste, produciendo en el oyente una sensación de altura o tono (tonos altos y bajos).

Como en la práctica es virtualmente imposible escuchar un tono puro, equivalente a una señal sinusoidal única y de una frecuencia determinada, a no ser que se genere artificialmente, se hace necesario contar con la tercera característica mencionada, el timbre, que da una idea del número de frecuencias diferentes, con niveles más o menos elevados que acompañan a la fundamental. En muchas ocasiones se realiza la mezcla de diferentes frecuencias, de forma consciente, para obtener sensaciones agradables, como es el caso de la música.

## 1.2. PROPAGACION DEL SONIDO

Una vez definidas las características de los sonidos, conviene repasar brevemente su forma de propagación hasta el oyente. Cualquier sonido como ya es conocido, se genera por la vibración de cierto elemento a una frecuencia superior a los 20 Hz e inferior a los 20 KHz.

Este movimiento produce unas compresiones

ones y expansiones en las capas de aire que le rodean, que provocarán la entrada en acción de otros volúmenes de aire más alejados, transmitiéndose la vibración inicial en forma de ondas de presión que se van amortiguando con la distancia. La velocidad de alejamiento de la fuente de sonido de estas ondas es de 340 m/s. Sin embargo, el sonido no sólo se propaga en el aire, sino que también lo hace a través de todo medio que posea un mínimo de elasticidad. La velocidad en cada medio es distinta de la del aire y depende de sus propiedades físicas, aunque por regla general en los líquidos y sólidos se observa una propagación mucho más rápida que en el aire. Esto es lógico si tenemos en cuenta el mecanismo que entra en juego en la transmisión, formado por la vibración de las moléculas del medio de que se trate, en el aire o en los gases están más separadas que en los líquidos y sólidos, con lo que el tiempo entre choque y choque entre dos contiguas será mayor.

### 1.3. EL OIDO HUMANO

Después de conocer los aspectos fundamentales de la generación y propagación de ondas acústicas, es interesante completar la descripción en la parte relacionada con la percepción del sonido por los seres humanos, lo que se podría denominar como mecanismo auditivo, consistente en transmitir al cerebro la sensación obtenida por los diferentes niveles de presión sonora.

El sonido incide sobre el pabellón au-

ditivo u oreja que hace el papel de bocina, concentrando la energía acústica recibida para enviarla al canal auditivo. Su forma, juega un papel relativamente importante respecto a la orientación, ya que gracias a ella se detecta con un cierto grado de aproximación la dirección por la que nos llegan las señales.

Las diferencias de presión alcanzan el tímpano, situado al final del canal auditivo, formado por una membrana que separa a éste del oído medio. Su superficie es de alrededor de 1 centímetro cuadrado. El efecto producido es su inmediata entrada en vibración al ritmo impuesto por la señal que llega, transmitiéndose al oído interno a través de una cadena de huesecillos denominados martillo, yunque y estribo. La zona final del oído interno llamada membrana basilar, es sensible a las diferentes frecuencias y transmite la información obtenida al nervio auditivo, de donde finalmente se propaga hasta el cerebro.

#### 1.4. FACTORES DE LA AUDICION

La audición consta de cierto número de procesos distintos, cuya complicación no permite encontrar una respuesta simple y única entre la medida física de la presión sonora y su percepción por el oído.

Debido a esto existe una falta de linealidad en la respuesta de éste, tanto en frecuencia como en amplitud. Se han realizado diversos experimentos para poner de manifiesto estas alinealidades.

Una de las medidas que se han realiza-

do, es la de la capacidad auditiva mínima a una determinada frecuencia, es decir, la intensidad mínima que puede ser oída a una frecuencia concreta.

Los resultados varían de forma considerable de unos individuos a otros.

En definitiva, para oír sonidos de distintas frecuencias con la misma sensación de sonoridad o de intensidad sonora, deben ser emitidos con distinto nivel de potencia, ya que el oído no responde linealmente, como ya se indicó anteriormente.

Por ejemplo, un sonido de 20 dB a una frecuencia de 1 KHz es perfectamente audible, mientras que otro con la misma intensidad sonora, pero con una frecuencia de 100 Hz está por debajo del umbral audible. La sonoridad, entendida como capacidad subjetiva de percibir los sonidos, depende no sólo de la intensidad, sino también de la frecuencia. Ya en 1933 Fletcher y Munsen experimentaron sobre hechos cuyos resultados se resumen en las curvas de igual sonoridad. Estas curvas se obtienen comparando dos sonidos puros, uno de referencia de 1KHz y otro de cualquier frecuencia, ajustando el nivel de intensidad del segundo para que su sonoridad subjetiva sea igual que la de 1KHz. Estas curvas demuestran que la mejor percepción sonora está entre los 600Hz y 10 KHz, aproximadamente.

Los resultados de estos experimentos son tenidos en cuenta en el diseño de amplificadores de audio, mediante el circuito de compensación fisiológica que produce un realce de las frecuencias altas y bajas, para ecualizar la escucha de estos sonidos respecto a las frecuencias medias, so-

En la presente tabla se detallan los valores de las frecuencias en las diferentes notas de la escala natural y de la temperada, en función de una hipotética frecuencia  $f$ , correspondiente a la nota DO.

Escala natural		Escala temperada		Ejemplo (frec. en Hz)	
Notas	Frecuencias	Notas	Frecuencias	Notas	Frecuencias
DO	$f$	DO	$f$	261	261,6
DO# RE $\flat$	$1,0417 f$ $1,08 f$	DO#	$1,0595 f$	271,9 281,9	277,2
RE	$\frac{27}{24} f = 1,125 f$	RE	$1,1225 f$	293,6	293,7
RE# MI $\flat$	$1,1719 f$ $1,2 f$	RE#	$1,1892 f$	305,9 313,2	311,1
MI	$\frac{30}{24} f = 1,25 f$	MI	$1,2599 f$	326,3	329,6
FA	$\frac{32}{24} f = 1,3333 f$	FA	$1,3348 f$	348	349,2
FA# SOL $\flat$	$1,3889 f$ $1,44 f$	FA#	$1,4142 f$	362,5 375,8	370
SOL	$\frac{36}{24} f = 1,5 f$	SOL	$1,4983 f$	391,5	392
SOL# LA $\flat$	$1,5625 f$ $1,6 f$	SOL#	$1,5874 f$	407,8 417,6	415,3
LA	$\frac{40}{24} f = 1,6667 f$	LA	$1,6818 f$	435	440
LA# SI $\flat$	$1,7361 f$ $1,8 f$	LA#	$1,7818 f$	453,1 469,8	466,2
SI	$\frac{45}{24} f = 1,875 f$	SI	$1,8877 f$	489,4	493,9
DO'	$2 f$	DO'	$2 f$	522	523,3

bre todo a bajos niveles de intensidad sonora en -- los que la diferencia de percepción es más acusada.

Existe además un fenómeno de pérdida - de audición en las personas, que se incrementa con la edad, siendo más acusado en los hombres que en las mujeres. Así, de la frecuencia máxima de 20.000 Hz que pueden percibir los jóvenes se pasa a 15.000 Hz para personas de cuarenta ca cincuenta años y a edades superiores puede quedarse reducida a tan sólo 10.000 Hz.

#### 1.5. LAS NOTAS MUSICALES

Cualquier melodía sea cual sea su forma de interpretación, se compone de una secuencia - de sonidos denominados notas, caracterizados por su frecuencia, entre los que existe una sucesión de pe ríodos de silencio más o menos largos.

Las frecuencias empleadas por estas no tas esán comprendidas entre los 27 Hz y los 4'2 KHz aunque algunos armónicos que las acompañan pueden - alcanzar valores muy altos, llegando a la proximi-- dad de 20 KHz.

Las notas musicales son siete y se de- nominan = DO, RE, MI, FA, SOL, LA, SI, y al conjun- to completo se le llama escala.

La relación de frecuencias entre ellas está perfectamente definida. Si la primera nota (DO) tiene una frecuencia  $f$ , las siguientes tienen valo- res de  $1'125 f$ ,  $1'25 f$ ,  $1'333 f$ ,  $1'5 f$ ,  $1'666 f$ , --  $1'875 f$  y a continuación aparecería la  $2f$  que es nu evamente la nota DO de la escala siguiente. Es de--



cir, que a partir de una escala se crea todo el espectro musical, repitiéndose hacia arriba y hacia abajo las notas con la relación indicada.

Los intervalos de separación de frecuencias entre cada dos notas no son todos iguales, siendo algunos bastante grandes. Para cubrir estas separaciones se crearon los semitonos, denominados bemol y sostenido e indicados con los símbolos b y # respectivamente.

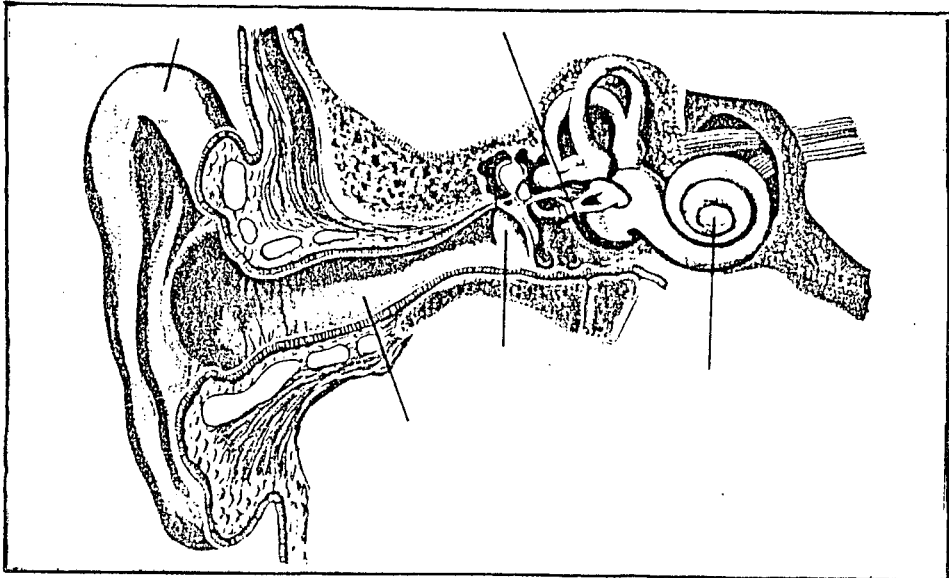
Si se considera una nota de frecuencia  $f$  su sostenido tiene un valor de  $\frac{25}{24} f$  y su bemol de  $\frac{24}{25} f$ .

Se entiende por octava el espectro de frecuencias comprendido entre una frecuencia  $f$  y el doble de la misma  $2f$ . Así puede observarse que en la escala natural obtenida con todas las notas más sus sostenidos y bemoles se incluyen 17 por octava, ya que ni la MI ni la SI admiten sostenidos, sucediendo lo mismo con los bemoles de las notas DO y FA.

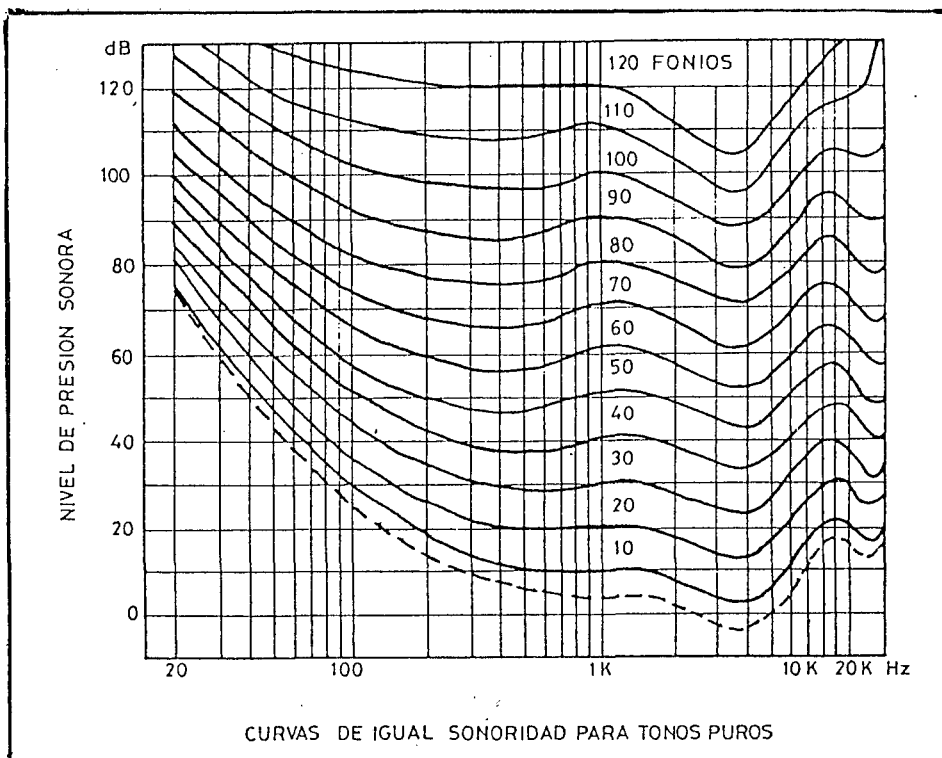
## 1.6. LA ESCALA TEMPERADA

Existe otra escala musical denominada escala temperada en la que el sostenido de una nota y el bemol de la siguiente coinciden, reduciéndose el número de notas a 12, evitándose así la elevadísima cantidad de teclas que poseerla, por ejemplo, un piano de 7 octavas.

Los teclados construidos para la escala temperada se dividen en dos grupos identificados por el color de las teclas, que son en la mayoría de los casos, blancas y negras, correspondiendo las



Representación de una sección transversal del oído humano, mostrando las diversas partes que la forman.



Curvas de Fletcher-Munson, denominadas de igual sonoridad. En ellas se observa la diferente sensibilidad del oído a tonos de distintas frecuencias.

primeras a las notas y las segundas a los sostenidos. Observando un teclado completo se verá que comienza con un grupo de siete teclas blancas, entre las tres primeras existen intercaladas dos negras y entre las cuatro últimas otras tres teclas negras.

Esta secuencia se repite a lo largo de todo el teclado.

La relación entre cada dos notas de la escala temperada es de  $1'05946$ , difiriendo los valores de sus frecuencias ligeramente de los de la escala natural aunque se conserva el mismo margen para el conjunto total.

La forma de deducir todas las frecuencias correspondientes a las notas, es tomar una de ellas como referencia. A finales del siglo pasado se tomó la nota  $LA_3$  para este efecto, cuya frecuencia es de 435Hz, aunque más tarde, en 1953 se redondeo su valor a 440 Hz.

La totalidad del espectro audible (20 Hz a 20 KHz ) contiene prácticamente 10 octavas considerando la nota más baja  $MI_{-1}$  (20'6 Hz) y la más alta  $RE_9$  (18'799 Hz). La octava indicada con -1 es la más grave y la 8 es la más aguda, apareciendo a continuación tres notas de la octava 9. Entre estos extremos se encuentran las denominadas 0, 1, 2, 3, etc.

## 137. REGISTROS

Además de la nota que define la frecuencia fundamental de un sonido, existen una serie de armónicos de diferentes frecuencias y una cierta

modulación del nivel de la misma, que caracterizan a cada uno de los instrumentos musicales y que permiten distinguir a un violín de un piano o a éste de una guitarra o una flauta, etc. En base a esto se denomina registro a todo el conjunto completo de sonidos que configuran a un determinado instrumento.

Un registro de trompeta, por lo tanto, será una señal eléctrica o acústica, que contenga la totalidad de frecuencias que produce dicho instrumento, cuando es accionado por su intérprete y lo mismo podría decirse de la totalidad de instrumentos conocidos.

## CAPITULO II

### 2.1. ANALIZADOR DE ESPECTRO AUDIO

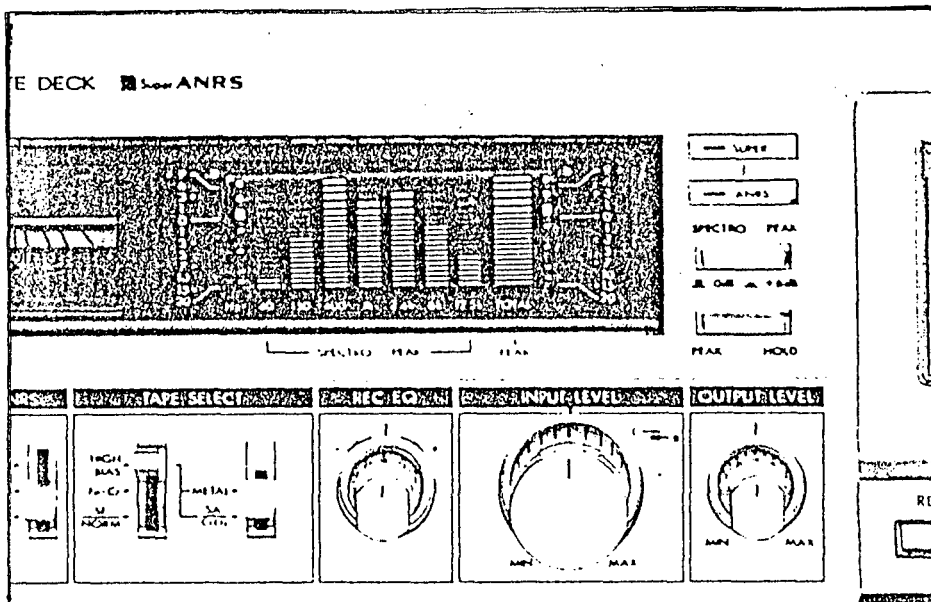
Cuando se intenta sonorizar una sala, ya sea una discoteca, un cine o la habitación familiar de escucha de programas musicales, lo primero que debe tenerse en cuenta es que dicha sala presente una respuesta lineal a todas las frecuencias audibles, de forma que no haya valores que resulten - acentuados y otros atenuados.

La misión básica de un ecualizador consiste en modificar ligeramente la respuesta en frecuencia del equipo de audio para compensar las deficiencias que, inevitablemente, toda sala de audición presenta, de forma que la respuesta del conjunto amplificador-sala, si sea lineal.

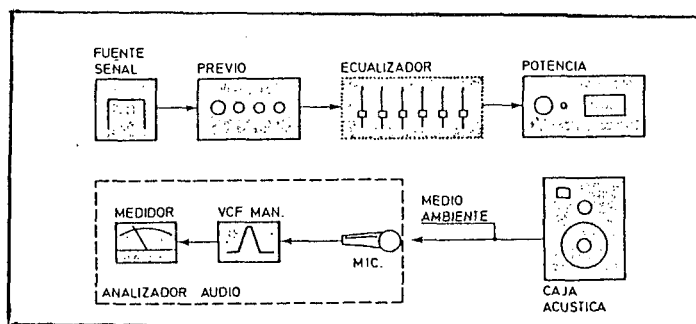
Si al equipo de audio se le hace reproducir una señal, cuyo espectro de frecuencia sea -- perfectamente conocido, por otro lado se obtiene la forma real en que dicha señal puede ser escuchada - en la sala de audición, podrán introducirse las correcciones necesarias. La misión de un analizador de audio es precisamente la de obtener tal respuesta de frecuencias.

### 2.2. ANALIZADORES EN TIEMPO REAL

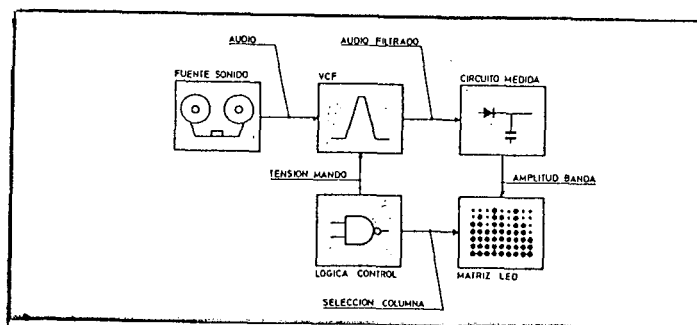
Un analizador que permite conocer en - cada momento la respuesta en frecuencia de una sala en todo el espectro de audio se denomina de tiempo real. Son los analizadores ideales, aunque también



Analizador de espectro integrado en un equipo de audio.



Esquema de principio de una medida de respuesta acústica.



Principio de funcionamiento de un analizador de audio compartido.

son los más complejos y caros.

Los más complejos utilizan técnicas osciloscópicas para hacer visible la respuesta en frecuencia. Sobre el canal horizontal se llevan los valores de las distintas frecuencias (habitualmente, de 20 Hz a 20kHz), mientras que sobre el vertical se muestran las amplitudes de señal correspondientes. Así, puede compararse el espectro original con el recogido, y extraer las diferencias.

En otras ocasiones, el analizador divide el espectro completo de frecuencias en pequeñas bandas (de una octava, o menores) y promedia el valor de las amplitudes de señal en ellas. Así se obtiene una visión razonablemente buena de la respuesta. Cuanto más estrechas sean las bandas, con mayor precisión se obtendrá la respuesta.

Este tipo de analizador suele incorporarse en algunos equipos amplificadores. Una serie de filtros divide el espectro de audio en 10 ó más bandas, mostrándose la amplitud relativa de cada banda sobre una columna de diodos luminosos (LED).

Cuanto mayor sea el número de ellos encendido, mayor amplitud de señal existe en la banda de frecuencia representada.

### 2.3. FILTROS COMPARTIDOS

La circuitería necesaria para el funcionamiento de un analizador es harto compleja, incluso en el caso de emplear filtros de banda. Gracias al empleo de filtros controlados por tensión (VCF),

construidos sobre la base de uno o varios operacionales del tipo OTA, pueden realizarse filtros pasabanda cuya frecuencia central se controla por medio de una tensión de mando.

Así, pueden construirse analizadores que incluyen la circuitería correspondiente a un solo filtro. En un analizador de este tipo, una determinada tensión de mando se aplica sobre el VCF, con lo cual se sintoniza sobre una cierta frecuencia.

El resultado se lleva a una columna de diodos luminiscentes, que mostrará la amplitud de señal a esa frecuencia. Instantes después, la tensión de mando se cambia, con lo cual el VCF es sintonizado sobre una frecuencia distinta. La misma tensión de mando gobierna una serie de conmutadores que hacen que el resultado de la nueva medida sea llevado a una columna luminiscente distinta.

El proceso se repite, de forma que el filtro es sintonizado sucesivamente sobre cada una de las frecuencias preeterminadas, y el resultado es llevado al indicador luminoso correspondiente.

Si el tiempo asignado a cada una de las lecturas es suficientemente pequeño, el resultado será una medida equivalente a la conseguida por un analizador que trabaje en tiempo real, pero con un considerable ahorro de circuitería (¡ y de precio!).

#### 2.4. ANALIZADORES ESTATICOS

Finalmente, cabe mencionar los analizadores que emplean un solo filtro que puede ser sin-

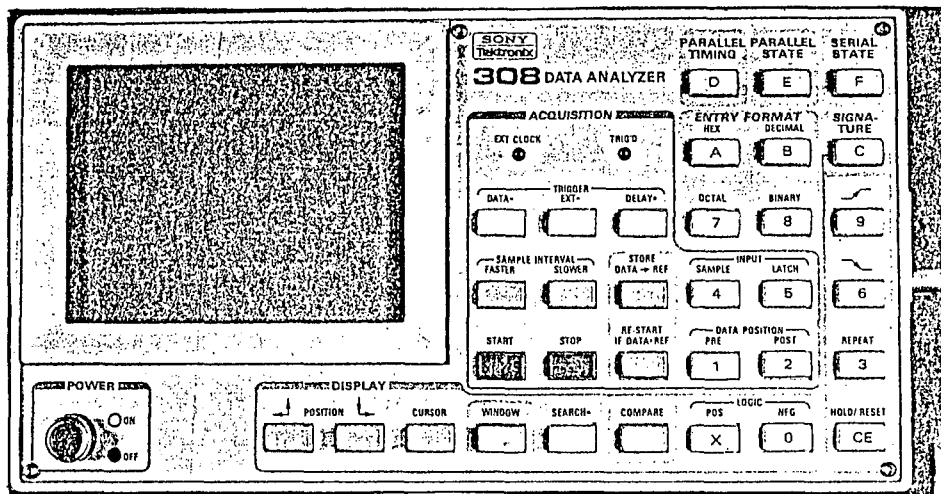


tonizado manualmente, leyéndose el resultado de la medida sobre una columna de diodos, o sobre un instrumento de aguja. Aquí, el proceso consiste en hacer que el amplificador reproduzca un cierto sonido bien que cubra el espectro completo de audio, o bien sólo aquella banda que va a tratarse por el analizador. Así se obtienen medidas sucesivas que, una vez compuestas, dan imagen de la respuesta global - que trata de conseguirse.

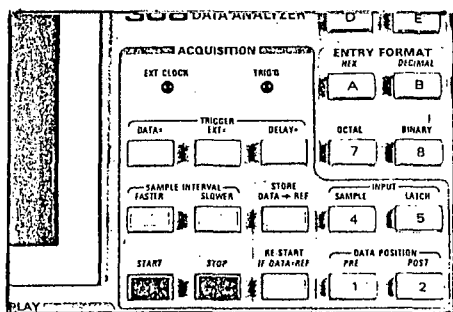
## 2.5. ANALIZADORES LOGICOS

La rápida evolución tecnológica de la electrónica en los últimos años y su tendencia hacia el trasvase de un gran número de funciones convencionalmente analógicas hacia técnicas digitales, ha producido un efecto importante en los sistemas clásicos de instrumentación, ya que cuando se pretendió analizar un equipo en el que existían varias vías - simultáneas de transferencia de información digital no resultó válido ninguno de los instrumentos conocidos hasta el momento, lo que originó el diseño y posterior desarrollo de los analizadores lógicos.

Este tipo de instrumentos se crearon -- con el propósito de resolver con un solo aparato el mayor porcentaje posible de los casos de análisis y detección de anomalías que pueden presentarse en la práctica, intentando evitar la necesidad de tener que disponer de varios instrumentos conectados simultáneamente al equipo bajo prueba. De esta manera no sólo se persigue la identificación de los fallos de funcionamiento, sino también el origen de los --



Aspecto de un analizador lógico de un modelo convencional, en el que todos los controles se efectúan por teclado.



Detalle de los controles de adquisición de datos, entre los que se incluye la selección del formato de entrada.

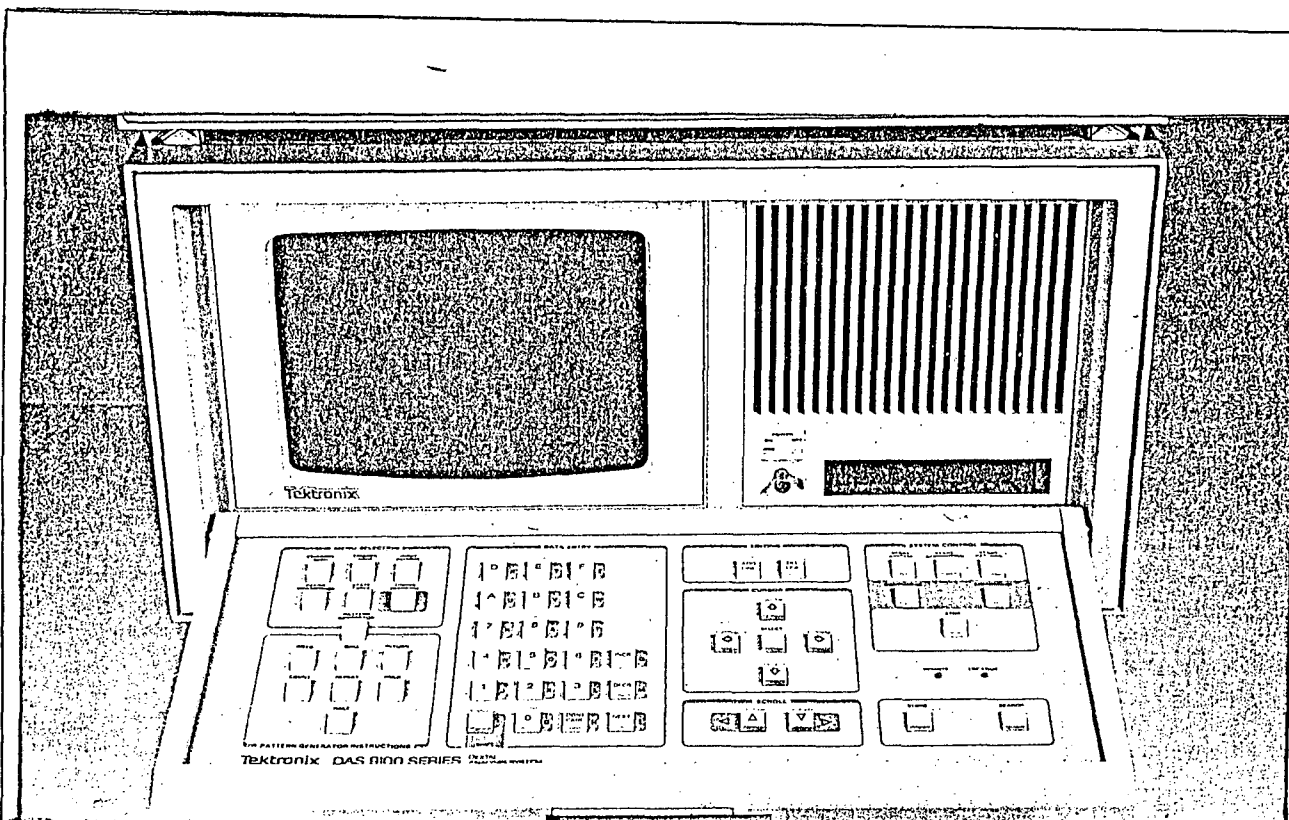
mismos.

Los analizadores lógicos de la primera época, consistían básicamente en analizadores de estados lógicos y disponían únicamente de un display o visualizador alfanumérico, en el que representaban en el sistema hexadecimal, una serie de estados sucesivos tomados de los "buses" de datos o de direcciones del sistema digital sometido a prueba, y que habían sido previamente almacenados en la memoria interna del sistema. Este equipo evolucionó rápidamente hasta los modelos actuales, en los que posee como mínimo dos posibilidades de trabajo: análisis de estados y visualización de los diagramas de tiempo simultáneos de todas las vías de los "buses".

El aspecto externo de estos equipos es bastante similar al de un osciloscopio, aunque con un mayor número de controles externos, efectuándose la presentación tanto de estados como de las formas de onda de los "buses", mediante el tubo de rayos catódicos normalmente utilizado por este instrumento. El funcionamiento interno del analizador está gobernado por un microprocesador, lo que permite una gran flexibilidad en el uso del mismo.

Cualquier analizador lógico de este tipo, se compone de cuatro secciones básicas: adquisición de datos de medida, memoria o almacén de datos sección de control y presentación sobre pantalla.

La sección de adquisición o toma de datos los recibe en paralelo y simultáneamente del equipo bajo prueba, mediante una serie de sondas especiales, situadas en los puntos adecuados. El circuito de entrada y las sondas mismas deben de tener



Analizador lógico de la última generación cuyo sistema de trabajo está basado en el procedimiento de "menú".

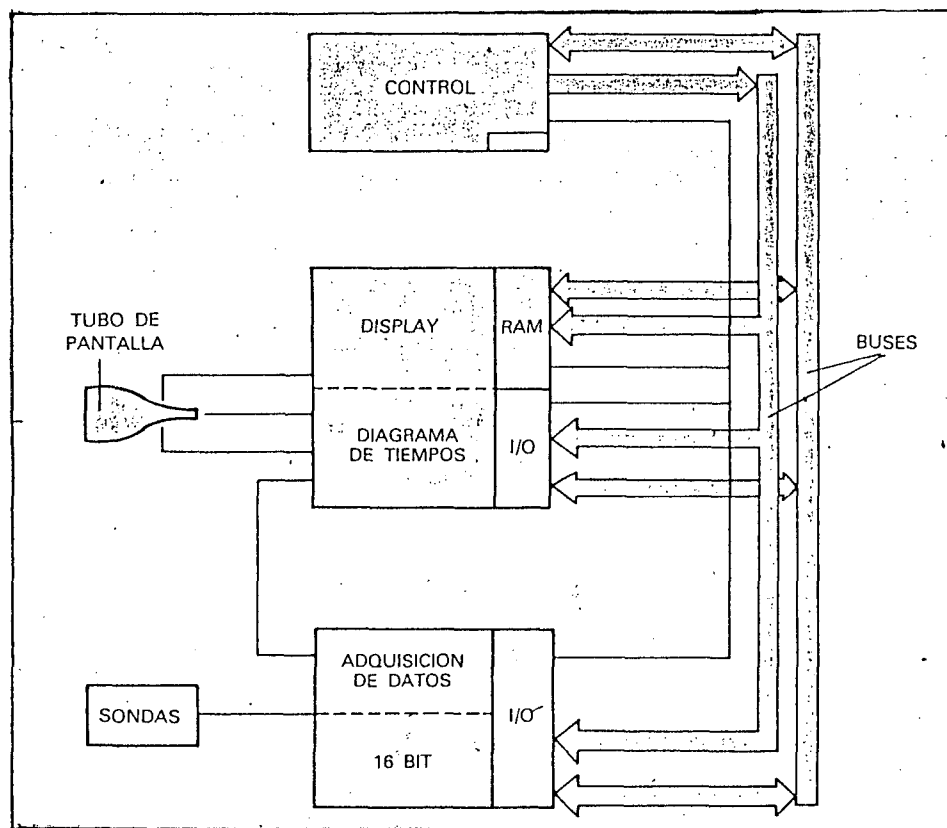


Diagrama de bloques de un analizador lógico.

unas características tales que no interfieran con el funcionamiento normal del equipo, ya que de lo contrario se obtendrían conclusiones falsas y podría incluso dañarsele. Ello implica el disponer de una elevada impedancia de entrada y que la propia sonda sea muy ligera, utilizándose en muchos casos un código de colores en las mismas o en los cables que las enlazan al equipo para poder identificarlas con facilidad.

Otra de las funciones necesarias en esta sección de toma de datos es el disponer de un nivel lógico ajustable para poder adaptarse a las diferentes familias de circuitos integrados que se encuentran en la práctica (TTL, ECL, NMOS, CMOS, etc.).

El número de canales de entrada que se precisa en la práctica varía en función de la utilización a que se destine el instrumento. Debido a ello se ha estandarizado en la gran mayoría de los analizadores un total de 16 canales, ya que se adaptan perfectamente a la gran mayoría de las aplicaciones y por otra parte representa ya un número bastante alto como para que la persona que lo utiliza pueda asimilar correctamente toda la información que aparece ante su vista. Existen dos procedimientos de adquisición de datos: síncrono y asíncrono.

El primero se emplea normalmente para efectuar el análisis de estados lógicos, representándose en la pantalla en sistema binario, octal o hexadecimal a voluntad del usuario, en este caso la secuencia de adquisición de datos del analizador depende de la frecuencia de reloj del equipo bajo prueba.

Algunos modelos incluyen dos o más posibilidades de adquisición de datos simultáneas.

El segundo procedimiento se utiliza para la representación de las formas de onda o diagramas de tiempos y en este caso la frecuencia a la que el instrumento toma los datos está determinada por su propio oscilador interno de reloj. El valor de frecuencia es muy importante cuando se desean "capturar" fenómenos transitorios o ruidos espúreos que causen alteraciones en el equipo, ello exige períodos de muestreo muy bajos del orden de 3 nanosegundos, lo que exige unas frecuencias de 300 a 400 MHz aproximadamente.

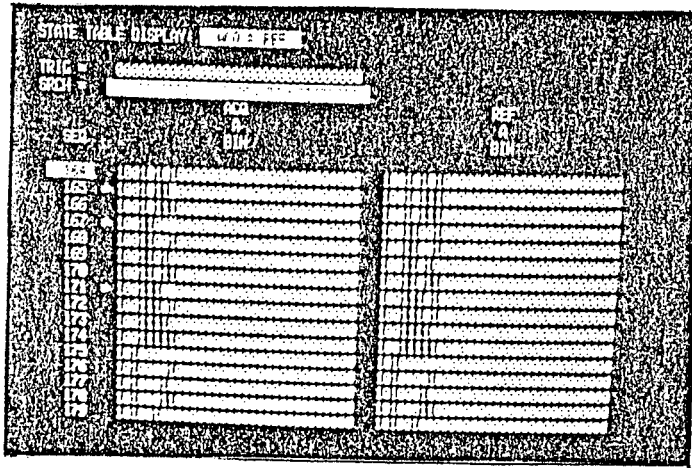
Algunos sistemas microcomputadores poseen un método de transmisión de datos multiplexado en el que tanto los bits como las direcciones de memoria se envían a través del mismo bus, mediante un circuito de división en el tiempo que asigna unos tiempos determinados de transmisión a cada uno de los dos "paquetes" de información. Esto exige al analizador lógico algún procedimiento de adquisición que asegure que los dos tipos de datos puedan ser recogidos y analizados, esto se consigue, en el modo síncrono, empleando dos señales de reloj dependientes de la del equipo analizado. La sección de almacenamiento de datos o memoria puede ser considerada como la más importante del sistema. Así, su constitución deberá ser lo suficientemente amplia como para poder acumular todos los datos simultáneos, tomados del exterior y lo bastante rápida como para que le puedan llegar los datos obtenidos a la elevada frecuencia de muestreo mencionada anterior-

mente. En realidad, aunque la frecuencia necesaria está definida por el sistema de operación asíncrono, suele estimarse en unas 10 veces superior a la del equipo bajo prueba.

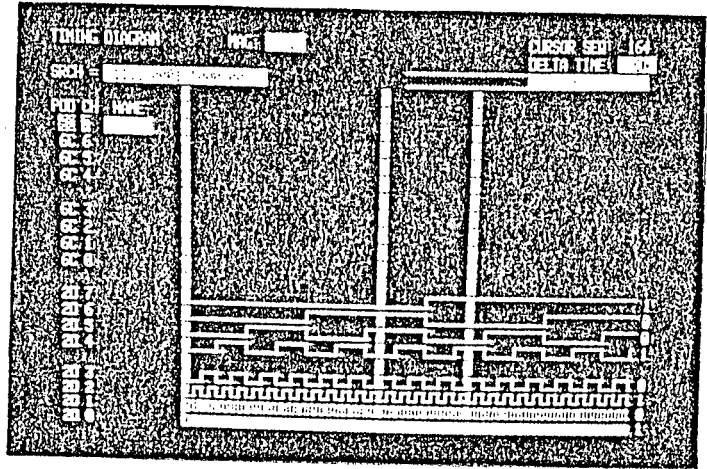
La capacidad de la memoria depende del modelo concreto de analizador de que se trate. En algunos casos puede llegar a 1K bytes agrupados en 512 palabras de 8 bits. Con esta posibilidad de almacenamiento pueden acumularse hasta 100 períodos de información a 20 MHz con una frecuencia de muestreo de 100 MHz.

La sección de control realiza como función principal la de disparo del sistema de adquisición de datos en el instante oportuno. El método que se emplea normalmente es el de identificación de una palabra determinada de información, en el que se compara los datos recogidos por el equipo con una palabra de 8 ó 16 bits fijada previamente por el usuario. En el instante en que aparece una coincidencia se produce el disparo. En algunos modelos se pueden emplear dos diferentes palabras, siendo este procedimiento aplicable en el caso en que puedan darse más de una condición para el disparo.

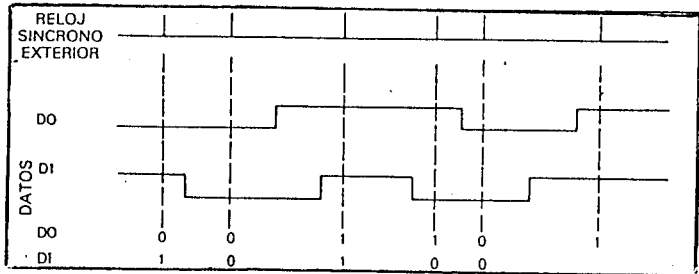
En el momento de producirse el disparo se detiene la entrada de datos al instrumento, sin embargo, algunos tipos de analizadores disponen de un cierto tiempo ajustable de retardo que permite anular la toma de datos en el instante que se desee con lo que podrá observarse en la pantalla toda la información precedente y posterior al momento en que una determinada palabra de información haya sido reconocida.



Representación de un diagrama de estados en código binario según el modo síncrono.

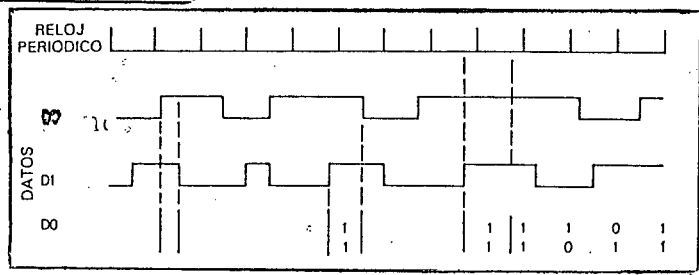


Representación de un diagrama de tiempos o forma de onda según el modo asíncrono.



Sistema de adquisición de datos según el modo síncrono.

Sistema de adquisición de datos en el modo asíncrono.





La sección encargada de la presentación de datos sobre la pantalla los recibe de la memoria y se encarga de llevar a cabo las dos posibles presentaciones = tabla de estados lógicos y diagramas de tiempos. La primera opción ofrecerá sobre la pantalla un listado de los datos recogidos por los 16 canales de entrada en binario, octal, hexadecimal y en algunos modelos se incluye la opción de presentación en el lenguaje ensamblador del microprocesador utilizado, empleando los nemónicos usuales de dicho lenguaje.

La presentación del diagrama de tiempos no suele ser la denominada de tiempo real, que es la correspondiente a las formas de ondas que realmente han aparecido en los buses, sino que presenta unas reconstrucciones de las mismas a base de los datos de niveles obtenidos durante el muestreo de las diferentes vías. Si el período de muestreo es bajo, se obtendrá una imagen muy similar a la real, pero en otras ocasiones pueden pasarse por alto la presentación de impulsos muy breves o de algunos ruidos o fenómenos transitorios también conocidos como "glitches".

En los modelos más modernos se ha sustituido el complejo panel de mandos que requiere un instrumento de este tipo por otro mucho más simple, denominado de "menú", en el que es propio instrumento el que pregunta al usuario cuál es la función que se desea realizar en cada momento, merced a un programa contenido en la memoria ROM interna. Estos equipos han sustituido el diseño de la caja exterior, como ya se ha indicado era similar a la de los

osciloscopios, por otros modelos que se asemejan a los empleados en sistemas informáticos, utilizándose además un tubo de rayos catódicos de deflexión electromagnética como los de los receptores de televisión. El último avance que se ha registrado en la presentación de los datos en pantalla ha sido la inclusión de tubos de color, los cuales facilitan la identificación de determinados detalles de la información ofrecida al usuario.

### CAPITULO III

#### 3.1. ANALIZADOR DE ESPECTRO DE ALTA TECNOLOGIA.- SU FUNCION.

El analizador de espectro es un instrumento relacionado con la medida de la frecuencia.

La función primordial de este equipo es la de separar las diferentes frecuencias o armónicos que contiene una determinada señal, analizando también la amplitud de cada uno, todo ello dentro de una banda previamente establecida. Así, por ejemplo, se se le aplica un tono o señal sinusoidal pura, el instrumento responderá que existe una única componente de frecuencia igual a la de la señal y cuya amplitud coincidirá exactamente con ella.

Si se aplica la señal de salida de un amplificador a cuya entrada se ha introducido la señal pura anterior, debido a la distorsión que inevitablemente produce este equipo, se obtendrán una serie de diferentes frecuencias perfectamente recogidos y presentados por el analizador. De igual forma podrá indicar cuál es el espectro de frecuencias de cualquier otra señal= instrumentos musicales, emisiones de radiodifusión, voces humanas, etc.

#### 3.2. PRESENTACION Y CONTROLES.

La presentación del resultado se realiza a base de un tubo de rayos catódicos en la que el eje horizontal representa valores de frecuencia y el eje vertical amplitudes.

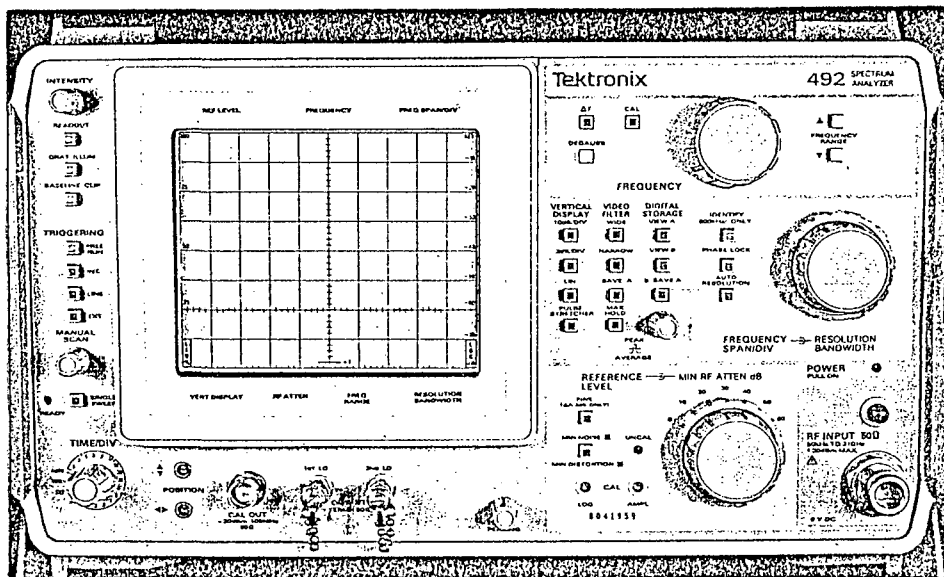
El instrumento presenta cinco controles básicos e indispensables para efectuar cualquier medida, así como otra serie de posibilidades complementarias que dependen del modelo de que se trate, estos controles son los siguientes:

.- Control de atenuación de entrada, - permite adaptar el equipo a una amplia variedad de niveles diferentes de entrada, de forma que todos ellos puedan ser representados en la pantalla.

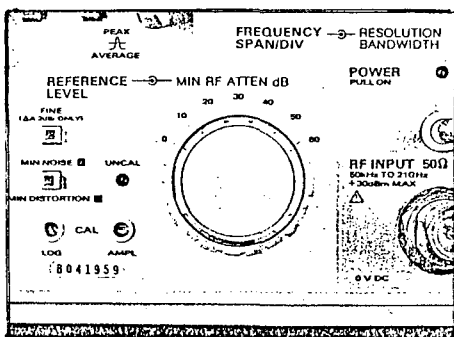
.- Selección de la frecuencia central, con él se realizará la búsqueda de los armónicos -- correspondientes a la señal aplicada, hasta que aparezcan representados en la pantalla. Normalmente se suele ajustar en un punto tal que el de mayor nivel o fundamental coincida con la línea vertical central de la retícula de medida, con objeto de poder determinar su frecuencia, leyéndola directamente sobre un dial calibrado o bien mediante la representación numérica de la misma en alguna zona de la pantalla.

.- Calibrado horizontal de la retícula con este control se elegirá el margen de frecuencias que corresponde a espacio entre dos líneas verticales de la retícula de la pantalla. Permite, por lo tanto, expandir más o menos la banda total de -- frecuencia representada. Además se utiliza también para medir la separación entre los diferentes armónicos y el fundamental o entre ellos, sin necesidad de tener que recurrir al control anterior y al dial.

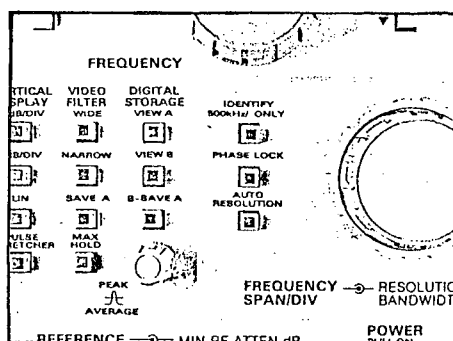
.- Calibrado vertical de la retícula, sirve para asignar la diferencia de nivel de señal que existe entre dos líneas horizontales de la re--



Analizador de espectro preparado para efectuar - medidas en alta frecuencia.



Sección de entrada del analizador. Puede verse el conector de entrada de señal y los mandos de atenuación y nivel de referencia.



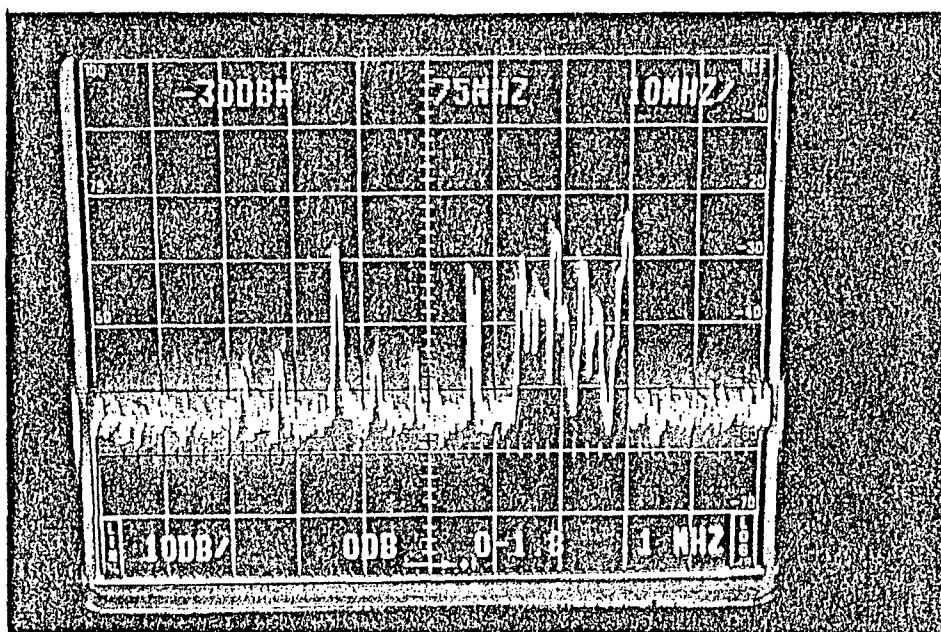
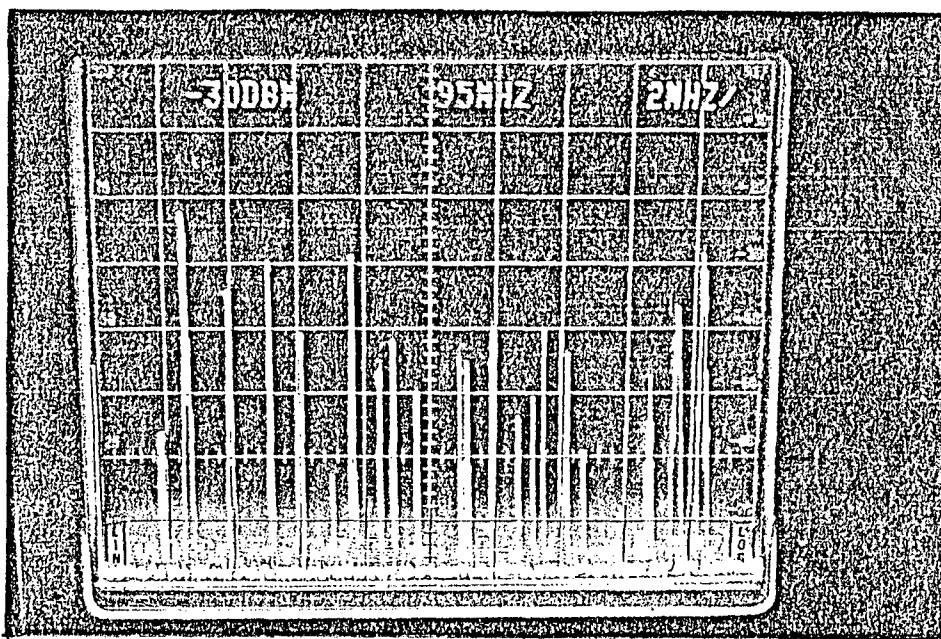
Controles del analizador de espectro destinados al calibrado horizontal, resolución y funciones complementarias.

tícula de la pantalla. Con él se podrá asignar la escala de medida más conveniente en cada caso y dependiente siempre de las características de amplitud de la señal aplicada así como de la precisión y exactitud que se desee tener en la medida. Normalmente existen dos posibilidades de calibrado, una de ellas es logarítmica con la retícula dividida en decibelios y la otra lineal, en cuyo caso la medida se efectuará en voltios, con una asignación fija de diferencia de tensión entre cada dos divisiones.

.- Nivel de referencia, indica el nivel tomado como referencia para la medida. Este factor es imprescindible debido a que el resultado se obtiene siempre en forma relativa al valor elegido con este control.

Otros controles complementarios que también tienen interés para el usuario son = el control de resolución de banda, el cual permite discernir la composición de un determinado armónico y dividirlo en otros subarmónicos más elementales, los filtros ajustables para eliminación de ruidos, el recortador de señal con el que se puede suprimir la presentación en pantalla de la zona de menor nivel de la señal y el control del disparo del barrido -- que efectúa el equipo.

La entrada de señal exterior suele contener una carga interna normalizada, con objeto de que la medida se efectúe en las condiciones normales de trabajo del equipo cuya salida se esté analizando, evitando así la aparición de reflexiones y ondas estacionarias en la línea de transmisión que enlaza los dos aparatos, las cuales redundarían en



Imágenes obtenidas en la pantalla del analizador de espectro correspondientes a la banda de FM - completa, en las que se pueden observar las portadoras de emisoras y un canal de TV.

una distorsión o inexactitud del resultado obtenido.

En la mayoría de los casos esta impedancia es de  $50\Omega$ , ya que coincide con la impedancia característica de los cables y equipos que trabajan en altas frecuencias. Si el instrumento está específicamente construido para atender a la banda de audio (20 Hz a 20 KHz), esta impedancia interna de carga tiene un valor de  $600\Omega$ .

### 3.3. CARACTERISTICAS NORMALES.

Las características normales de un analizador de espectro para radiofrecuencia son las siguientes:

.- Rango de frecuencias de medida, de 50 KHz a 1000 MHz (1 GHz).

.- Calibrado de frecuencia, de 10 KHz a 200 MHz por división.

.- Resolución de banda, de 1 KHz a 1 MHz seleccionable por décadas.

.- Nivel de referencia, de  $-120\text{ dBm}$  a  $+40\text{ dBm}$ .

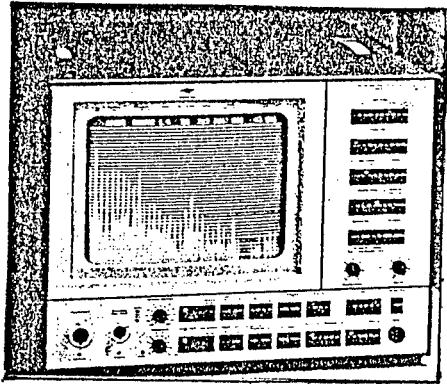
.- Calibrado de niveles, 10 dB/ div., 2dB/ div. y 8 divisiones lineales.

.- Atenuación de entrada, 30 dBm.

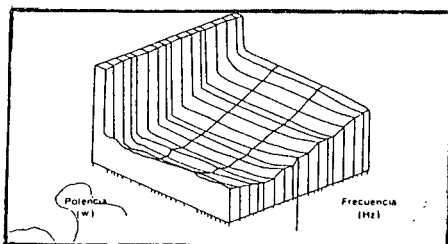
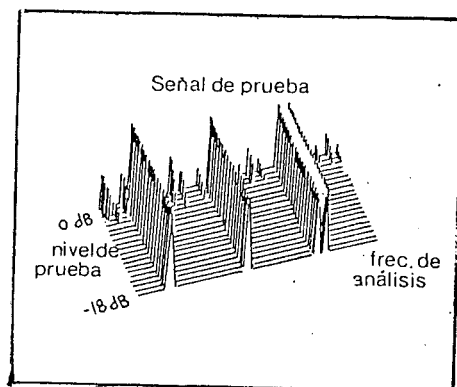
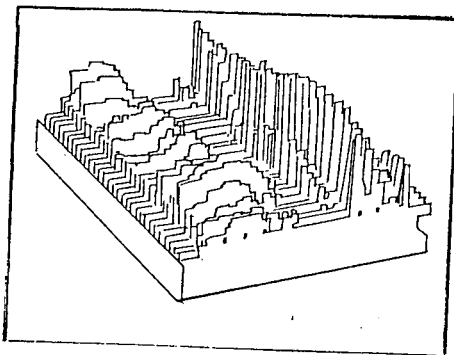
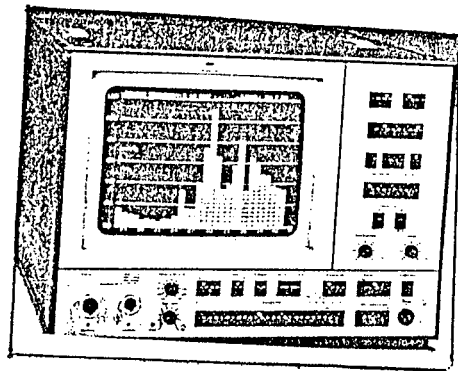
### 3.4. FORMAS DE REPRESENTACION

También existen los instrumentos denominados espectrógrafos, que en realidad son analizadores de espectro que muestran el resultado de la medida en forma de gráfica (curva de distribución,





Analizador de espectro  
para aplicaciones de  
audio.



Cada vez son más frecuen-  
tes los espectrogramas en  
tres dimensiones, que re-  
lacionan las variaciones

de tres parámetros simultáneamente.

estadística de valores, amplitud de las distintas componentes en función de la frecuencia, etc.).

Determinados equipos magnetofónicos y amplificadores de la gama alta incorporan medidores de señal (pico, medio o eficaz) en forma de varias líneas o barras de diodos LED o similares.

Realmente se trata de espectrómetros simplificados, que dan la amplitud de la señal sobre valores concretos de frecuencia.

Modernamente, y con la ayuda de técnicas de ordenador, pueden incluso obtenerse gráficos con tres dimensiones, en los que puede relacionarse por ejemplo, potencia, distorsión y respuesta en frecuencia de un amplificador en forma de una superficie (que es el equivalente en tres dimensiones a la línea o curva de variación en sólo dos dimensiones), u otras muchas combinaciones, como la variación que experimenta el espectro de frecuencia de una nota de un piano con el tiempo (ejes= tiempo frecuencia, amplitud).

EXPOSICION      TECNICA

## MONTAJE.- ANALIZADOR ESPECTRO DE AUDIO

### INTRODUCCION.-

Según expusimos en la primera parte de este proyecto el analizador de espectro de audio es un instrumento de medida que indica las componentes espectrales de una señal y sus respectivos niveles acústicos. Para ello descompone el espectro de audio en las denominadas bandas de armónicos, de un tercio de octava cada una.

El espectro de frecuencias audibles va de 20 a 20.000 Hz aproximadamente. Este margen de frecuencias es dividido por el analizador en 30 bandas, cada una de las cuales posee un ancho de  $1/3$  de octava. La frecuencia central de la banda más baja es 25 Hz y la de la más alta 20 KHz.

La función de este analizador aparte de realizar esta subdivisión, es la de visualizar el nivel de la señal contenida en cada una de las treinta bandas.

La visualización se realiza de dos formas distintas: una a través de una matriz de diodos LED's y otra por medio de un ordenador que recoge la señal del multiplexor convertida a digital gracias a un convertidor analógico-digital.

Este instrumento tiene muchas aplicaciones pero quizá la más destacable sea aquella que permite realizar el acondicionamiento acústico de locales y poder conseguir la sintonía más perfecta posible de un sistema estereofónico en cualquier zona de escucha con un ecualizador de  $1/3$ .

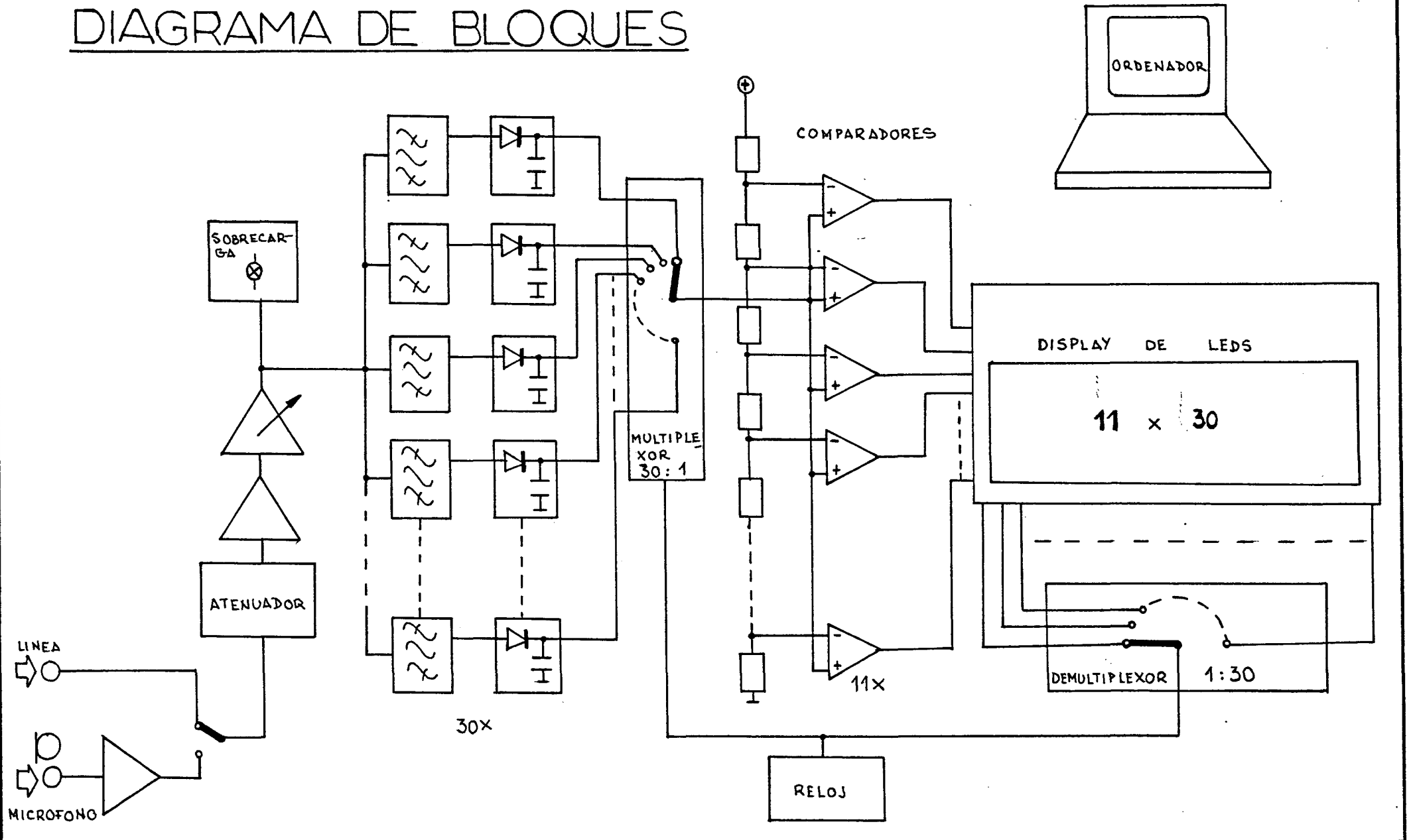
## 1.- DIAGRAMA DE BLOQUES.

Haremos una introducción muy general a poyándonos en el esquema de bloques que presentamos a continuación.

El circuito está provisto de dos entradas: una para las señales de línea y otra para las más débiles de un micrófono, que un amplificador se encargará de elevar al nivel de las de línea. A continuación encontramos un atenuador graduado en pasos de 10 dB, un amplificador de ganancia variable y los 30 filtros pasabanda, con frecuencias centrales esclonadas entre 25 Hz y 20 Hz. Cada uno de estos filtros alimenta a un rectificador de media onda, cuyas salidas están unidas al multiplexor de 30 a 1. La única señal resultante se compara con 11 --tensiones de referencia. Los once resultados de estas comparaciones controlan las filas de una matriz de diodos LED de 11 X 30. Las columnas están controladas por un demultiplexor de 1 a 30.

El reloj, común a ambos circuitos de -control (de filas y columnas), asegura una distribución síncrona de la información en la matriz de diodos LED. Cuando, por ejemplo, la salida del primer filtro está conectada al comparador a través del demultiplexor. Cuando es la salida del segundo filtro la que se conecta, se activará la segunda columna y así sucesivamente. Este analizador incluye una resolución variable de la visualización y un generador de ruido rosa que no aparecen indicados en el diagrama de bloques.

# DIAGRAMA DE BLOQUES



## 2.- FUENTE DE ALIMENTACION.-

La sección ó fuente de alimentación, - compuesta por un circuito o conjunto de ellos, es - siempre imprescindible en cualquier equipo electrónico. Su misión es la de proporcionar las tensiones de alimentación, generalmente continuas, al resto - del equipo. Se puede ver su esquema en la fig.(3).

### 2.1.- OPERACIONES PRINCIPALES.-

Cualquier fuente de alimentación, in-- corpora siempre un conjunto de funciones conside-- radas como imprescindibles, pudiéndose añadir a éstas, según los casos, otras que mejoren y com-- plementan el circuito básico.

Las operaciones que se deberán reali-- zar siempre, son las siguientes:

- .- Protección de sobretensiones o cor-- tocircuitos.

- .- Transformación de la tensión de en-- trada, obtenida en la red, a los niveles neces-- rios.

- .- Rectificación de las tensiones al-- ternas.

- .- Filtrado de la corriente continua - obtenida.

A estas se pueden añadir tres funcio-- nes más con las que se puede llegar a obtener -- una tensión y corriente continua tan perfecta co-- mo se desee:

- .- Filtrado adicional.

- .- Estabilizadór de tensión.

- .-

.- Autoprotección contra sobrecargas.

## 2.2.- ANALISIS APROXIMADO DEL FILTRO DE CONDESADOR.-

Por ser este tipo de filtro uno de los más utilizados, existen procedimientos analíticos y gráficos de su diseño. El método práctico más extendido para diseñar el filtro de condensador es un procedimiento gráfico que deberán utilizar en realidad. Su descripción aparece en algunos textos.

No obstante, debido a la finalidad de estas notas, creemos conveniente describir un análisis aproximado que, además de su sencillez, permite comprender fácilmente la problemática del diseño de este tipo de filtro.

Estudiaremos el circuito de la fig.(1) constituido por un rectificador de onda completa seguido del filtro condensador.

El funcionamiento cualitativo es el siguiente. El condensador, C, almacena energía durante el tiempo de conducción, entregándola a la carga durante el tiempo de no conducción. De este modo, aumenta el tiempo durante el cual la corriente circula por  $R_L$ , reduciéndose notablemente el rizado.

Si C no existiese, la tensión  $v_s$  tendría, como ya hemos visto, la forma indicada a puntos en la fig.(2).

La presencia de C da origen a la forma de onda representada con línea continua en la misma figura. Durante  $T_1$ , tiempo total de conducción, la tensión en el secundario del transformador



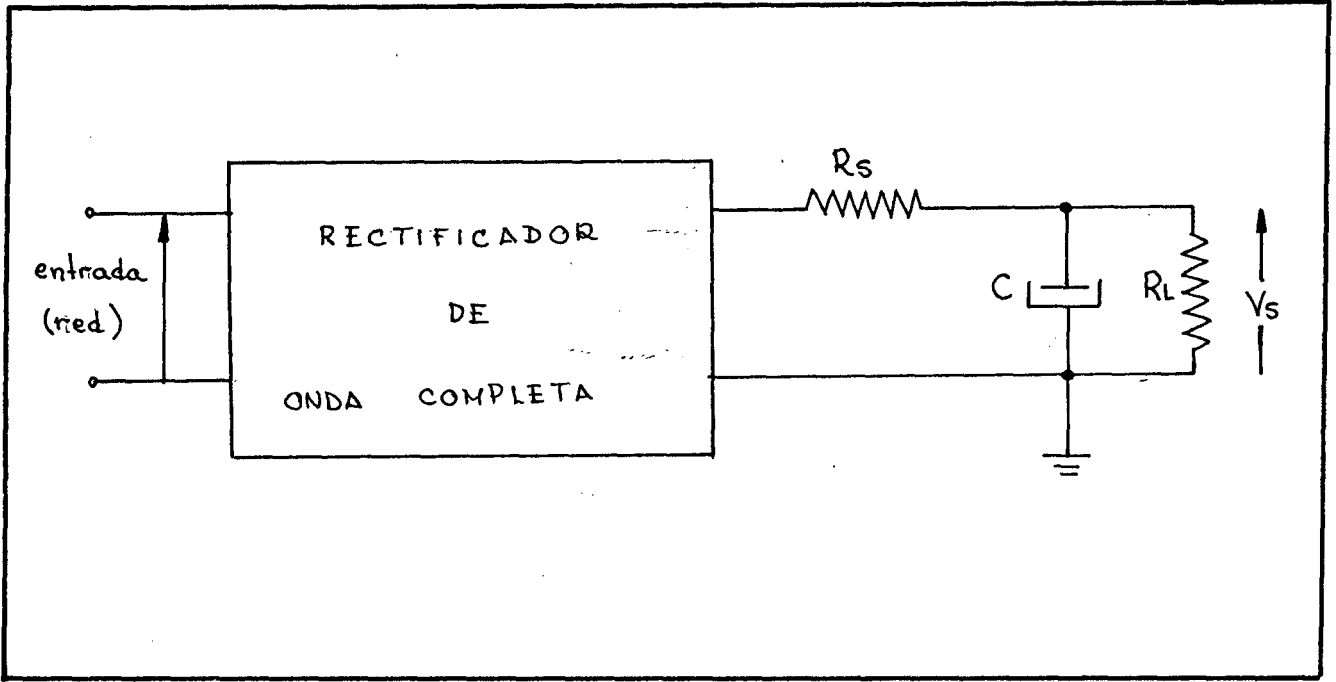


Fig. 1

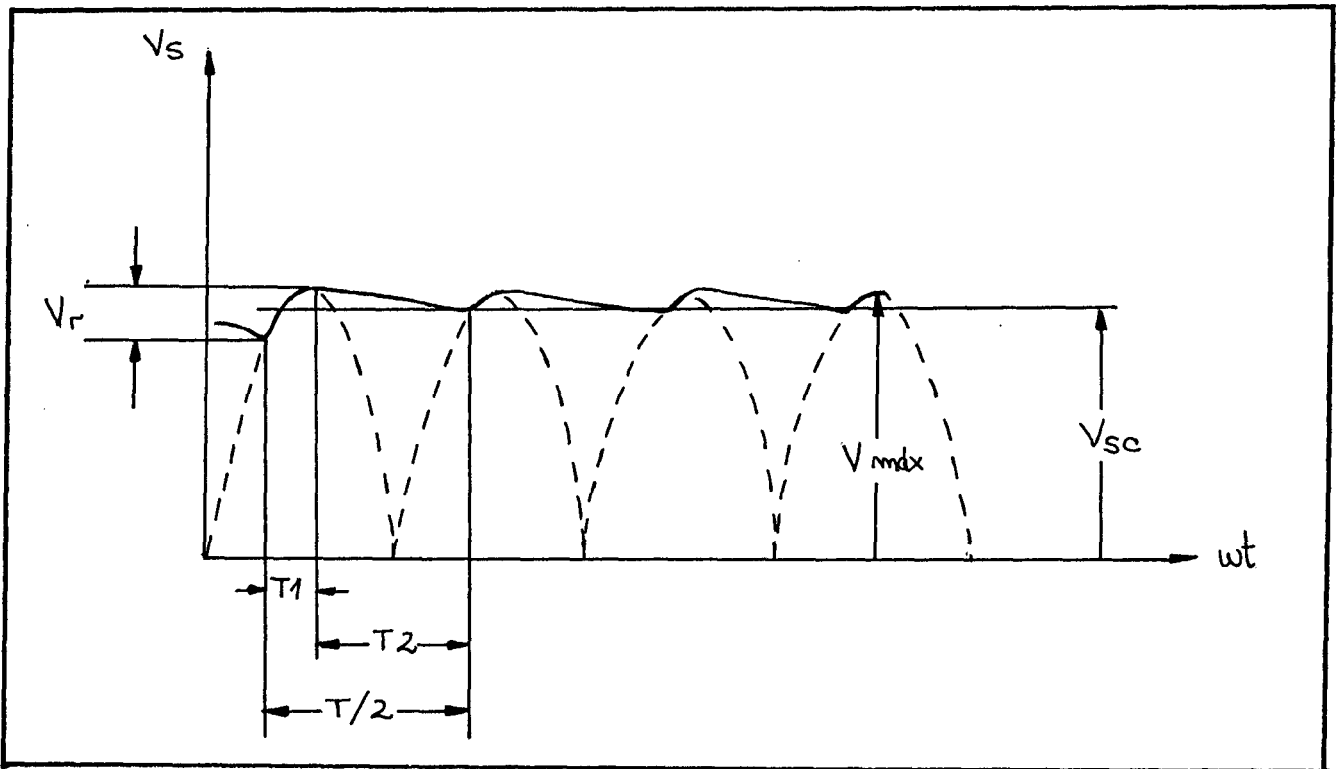


Fig. 2

dor excede a la tensión en bornas del condensador, polarizando el diodo del rectificador en directo, pasando el trozo de forma de onda dibujado. Durante  $T_2$ , tiempo total de no conducción del diodo, el condensador  $C$  se descarga sobre  $R_L$  ya que al predominar la tensión en el condensador sobre la del secundario el diodo queda bloqueado.

La aproximación que haremos es linealizar ambos tramos. Es decir, consideraremos que la forma de onda de  $v_s$  es triangular, con una amplitud de pico a pico de valor  $V_r$ .

En este supuesto, el valor medio (componente continua) de la tensión de salida es aproximadamente

$$V_{SC} = V_{m\acute{a}x} - \frac{V_r}{2} \quad (1)$$

Para hallar la curva de regulación hemos de expresar  $V_r$  en función de la corriente media en la carga,  $I_{SC}$ . Para ello, nótese que el condensador  $C$ , durante  $T_2$ , pierde una carga de valor  $I_{SC}T_2$ . De modo que la variación de tensión en el condensador,  $V_r$ , es igual a

$$V_r = \frac{I_{SC} T_2}{C} \quad (2)$$

Si la acción de filtrado es buena,  $T_1$  será pequeño frente a  $T_2$ , que se aproximará a  $T/2$ . Esto es

$$T_2 = \frac{T}{2} = \frac{1}{2f} \quad (3)$$

siendo  $f$  la frecuencia de la red (50 Hz). Entonces

$$V_r = \frac{I_{SC}}{2fC} \quad (4)$$

Resultando que

$$V_{SC} = V_{m\acute{a}x} - \frac{I_{SC}}{4fC} \quad (5)$$

expresión analítica, buscada, de la curva de regulación. Obsérvese de este resultado que el valor medio de la tensión de salida en vacío ( $I_{SC}=0$ ) ha aumentado de  $2 V_{m\acute{a}x}/\pi$  a  $V_{m\acute{a}x}$ . Asimismo, nótese que en este caso la resistencia efectiva de salida es igual a  $1/4fC$ . El rizado es directamente proporcional a  $I_{SC}$  e inversamente proporcional a  $C$ . Por tanto, para conseguir un bajo rizado y asegurar una buena regulación, deben utilizarse condensadores de gran capacidad (del orden de decenas de microfaradios).

Estos condensadores suelen ser electrolíticos.

Las ventajas de este tipo de filtro son:

- .- Pequeño rizado.
- .- Tensión de salida alta con cargas pequeñas ( Si  $I_{SC}=0$  ,  $V_{SC}= V_{m\acute{a}x}$  ).

Los inconvenientes son:

- .- Mala regulación y rizado alto con cargas grandes.
- .- Picos de corriente muy elevados que deben soportar los diodos durante  $T_1$ . Los picos serán más elevados conforme mayor sea  $C$ . Para reducir este inconveniente suele insertarse la resistencia limitadora,  $R_S$ , en serie, que a su vez limita la corriente de pico a través del rectificador en el instante inicial.

Para calcular el valor del rizado en este circuito es preciso hallar el valor eficaz de la onda triangular, sin  $V_{SC}$ . Si se hace esto resulta:

$$V_{\text{ef onda triangular}} = \frac{V_r}{2\sqrt{3}} \quad (6)$$

que sólo depende del valor de pico a pico y es independiente, por tanto, de las pendientes y de las longitudes de los tramos lineales.

$$r = \frac{V_r/2\sqrt{3}}{V_{SC}} = \frac{\overset{\text{Por tanto}}{I_{SC}}}{4 \cdot \sqrt{3} \cdot f \cdot C \cdot V_{SC}} = \frac{4}{4 \cdot \sqrt{3} \cdot f \cdot C \cdot R_L} \quad (7)$$

Los datos para un diseño suelen ser --  $V_{SC}$ ,  $I_{SC}$ ,  $r$  (%) y  $f$  (50 Hz). En primer lugar se determina el valor de  $R_L$ , teniendo en cuenta que:

$$R_L = \frac{V_{SC}}{I_{SC}} \quad (8)$$

De la Ec.(7) se calcula el valor de  $C$ ,  $R_S$ , suele tomarse del 1 al 10% de  $R_L$ . Resta sólo elegir el transformador y los diodos teniendo en cuenta las limitaciones previamente comentadas. Debido a los intensos picos de corriente, el transformador tendrá un comportamiento marcadamente no lineal de modo que su simulación mediante circuito equivalente no será válida. Este detalle ha de tenerse en cuenta a la hora de diseñar dicho transformador.

### 2.3.- FUNCION Y JUSTIFICACION DE COMPONENTES.-

La fuente de alimentación del analizador está provista de un fusible F1 que la protege contra los cortocircuitos y sobretensiones, incluido en la conexión a la red del primario del transformador Tr1. Este entrega en su secundario una tensión alterna, con un nivel en voltios adecuado para obtener en la salita la tensión continua adecuada.

Los cuatro diodos señalados con D5, D6 D7 y D8 se disponen en el montaje denominado puente

y producen una rectificación de onda completa. Se utiliza una toma central del transformador que se hace de tierra para obtener una alimentación simétrica.

El funcionamiento del conjunto de diodos es el siguiente durante un cierto intervalo de tiempo los diodos D5 y D8 conducen, quedando bloqueados D6 y D7, durante el siguiente intervalo se produce la situación opuesta conduciendo D6 y D7 y quedando cortados D5 y D8. De este modo a la salida -- del puente una tensión de 30 voltios.

La fuente consta de dos reguladores de tensión integrados, uno para tensiones positivas --- (7812) y el otro para tensiones negativas (7912) asegurando una alimentación simétrica estable de  $\pm 12$  voltios.

Debemos hacer notar que la fuente no regulada presenta un simple capacitor de filtro, lo que es característico para los reguladores de circuitos integrados actuales. Este capacitor siempre -- presenta un valor alto con el fin de eliminar el mayor rizado posible.

Los reguladores (7812) y (7912) presentan alrededor de 71dB y 60dB de rechazo de rizado -- lo que equivale a una atenuación de 3548 y 1000 respectivamente. Estos valores tan elevados hacen que ambos circuitos integrados no sólo actúen como reguladores sino también como filtros, lo que hace que dicho comportamiento contribuya a disminuir el valor del condensador de filtro abaratando, en consecuencia, el coste de la fuente.

Cuando el circuito integrado se encuentra a más de unas cuantas pulgadas del capacitor de filtro de la fuente, la inductancia de contacto puede producir oscilaciones dentro de circuito integrado debido a la realimentación por la fuente. Para evitar esto se pone a la entrada un condensador de  $0.33\mu\text{F}$ .

Para mejorar la respuesta transitoria del 7812 y 7912 se coloca un capacitor de salida -- que cortocircuite la corriente alterna cuyo valor -- característico es de  $0.1\mu\text{F}$ .

Los valores de estos condensadores los da el fabricante.

Empleando la fórmula del apartado anterior para el rizado, vamos a hallar el valor del condensador de filtro:

$$r = \frac{I_{SC}}{4 \cdot f \cdot \sqrt{3} \cdot C \cdot V_{SC}} \quad , \quad C = \frac{I_{SC}}{4 \cdot f \cdot \sqrt{3} \cdot r \cdot V_{SC}}$$

$$I_{SC} = 1 \text{ A.}$$

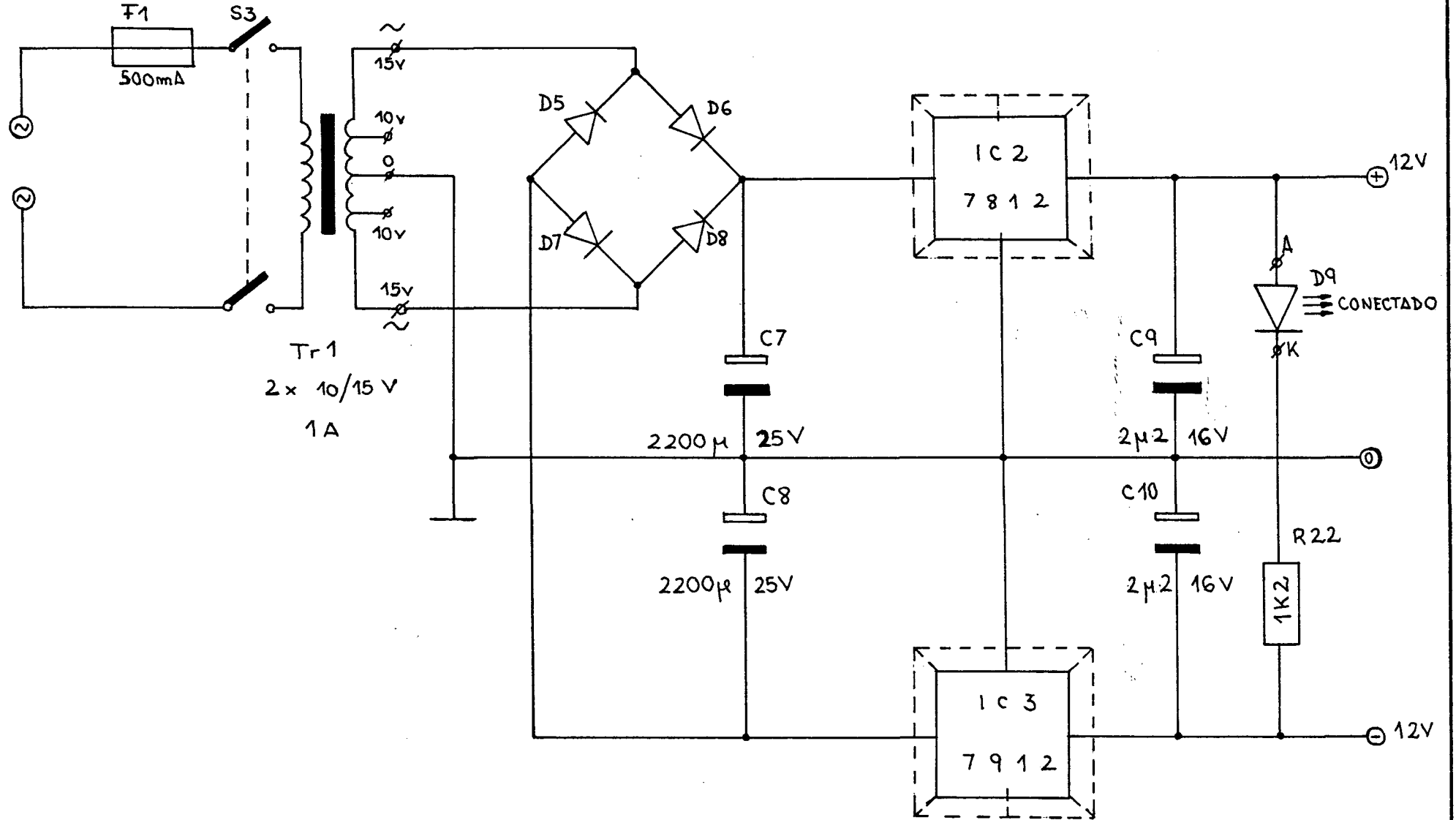
$$V_e \text{ (tensión a la entrada del regulador)} \\ = \frac{2.15 \cdot \sqrt{2}}{1} = 13.5 \text{ voltios.}$$

Tomado un factor de rizado del 9% para el condensador antes de poner el regulador tenemos:

$$C = \frac{1}{4 \cdot 50 \cdot \sqrt{3} \cdot 0.09 \cdot 13.5} = 2375 \mu\text{F.}$$

Se puede observar que hemos tomado un factor de rizado con lo cual el valor del capacitor ha bajado con relación a los valores característicos, pero si tenemos en cuenta la función de filtraje de los reguladores este rizado se atenuará unas mil veces más, minimizándolo y permitiendo el empleo

Fig. 3



o de una capacidad más baja.

La fuente de alimentación de la placa base es idéntica a ésta, con la excepción de que -- los reguladores han sido escogidos para proporcio-- nar una tensión continua de  $\pm 8$  voltios y la tensi-- ón suministrada desde el transformador es de 10v.c.a. a.

El diodo D9 indica cuando hay tensión a la salida.

### 3.- CIRCUITO DE ENTRADA.

El circuito de entrada del analizador-- presenta dos posibles entradas: la de línea y la -- del micrófono. La elección de cualquiera de ellas -- se realiza mediante el interruptor S1.

Veamos primero la entrada del micrófo-- no ( cuando S1 está en la posición A). El circuito corresponde a la fig. (4).

La impedancia de entrada del amplifica-- dor para microfónos, basado en A1, es de  $47\text{ K}\Omega$ , va-- lor adecuado para la mayor parte de los mismos. La ganancia se puede variar entre 46 y 76 dB con el em-- pleo de P1. Esta característica es modificable eli-- giendo adecuadamente el valor de R2.

Calcularemos a continuación la fórmula de la ganancia para obtener los valores de los de-- más componentes una vez conocidos el margen de ga-- nancia variable:

$$V_e = V_i + I \cdot R_3 \quad (1)$$

$$I = \frac{V_o}{R_2 + P_1 + R_3} \quad (2)$$



Por la teoría del amplificador operacional, la fórmula de la ganancia es:

$$A1 = \frac{V_o}{V_i}$$

de donde:

$$V_i = \frac{V_o}{A1} \quad (3)$$

Sustituyendo la ecuación (3) en la (1) queda:

$$V_e = \frac{V_o}{A1} + \frac{V_o}{R2 + P1 + R3} \cdot R3 \quad (4)$$

En la ecuación (3) se puede ver que A1 es muy grande con respecto a Vo con lo cual podemos despreciar este término en la ecuación (4), quedando de la siguiente forma:

$$V_e = \frac{V_o}{P1 + R2 + R3} \cdot R3 \quad (5)$$

Luego la expresión de la ganancia (Av) será:

$$A_v = \frac{R2 + P1 + R3}{R3} \quad (6)$$

Fijando el valor de R2 igual a 150K y P1= 100K, el valor de R3 sustituyendo en la ecuación (6) será:

$$46 = \frac{150K + 0 + R3}{R3}$$

Despejando la incógnita:

$$46 R3 = 150K + R3$$

ímplica:

$$45 R3 = 150K$$

de donde:

$$R3 = 150K / 45 = 3333 \cdot 3 \Omega = 3K3 \Omega$$

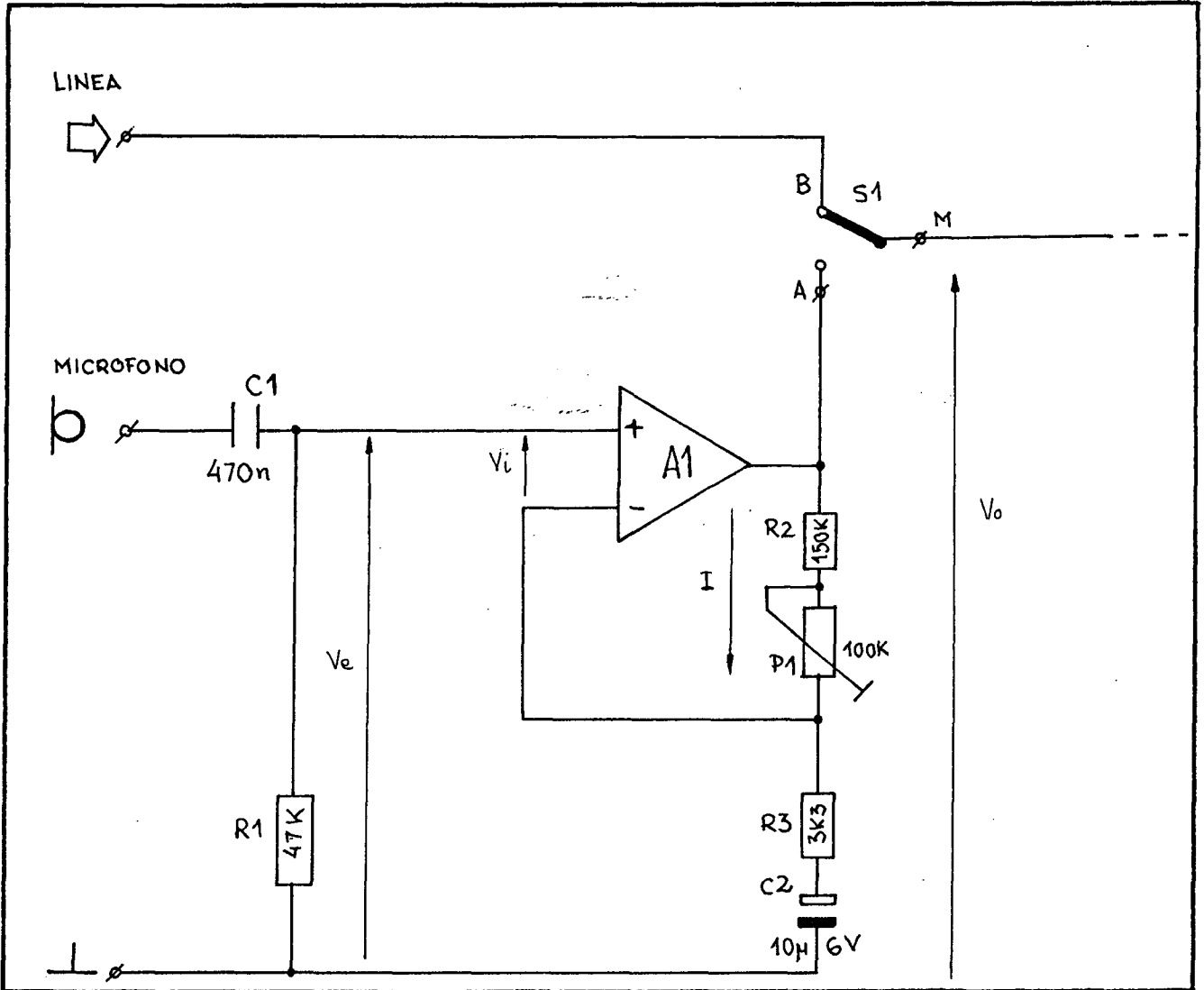


Fig. 4

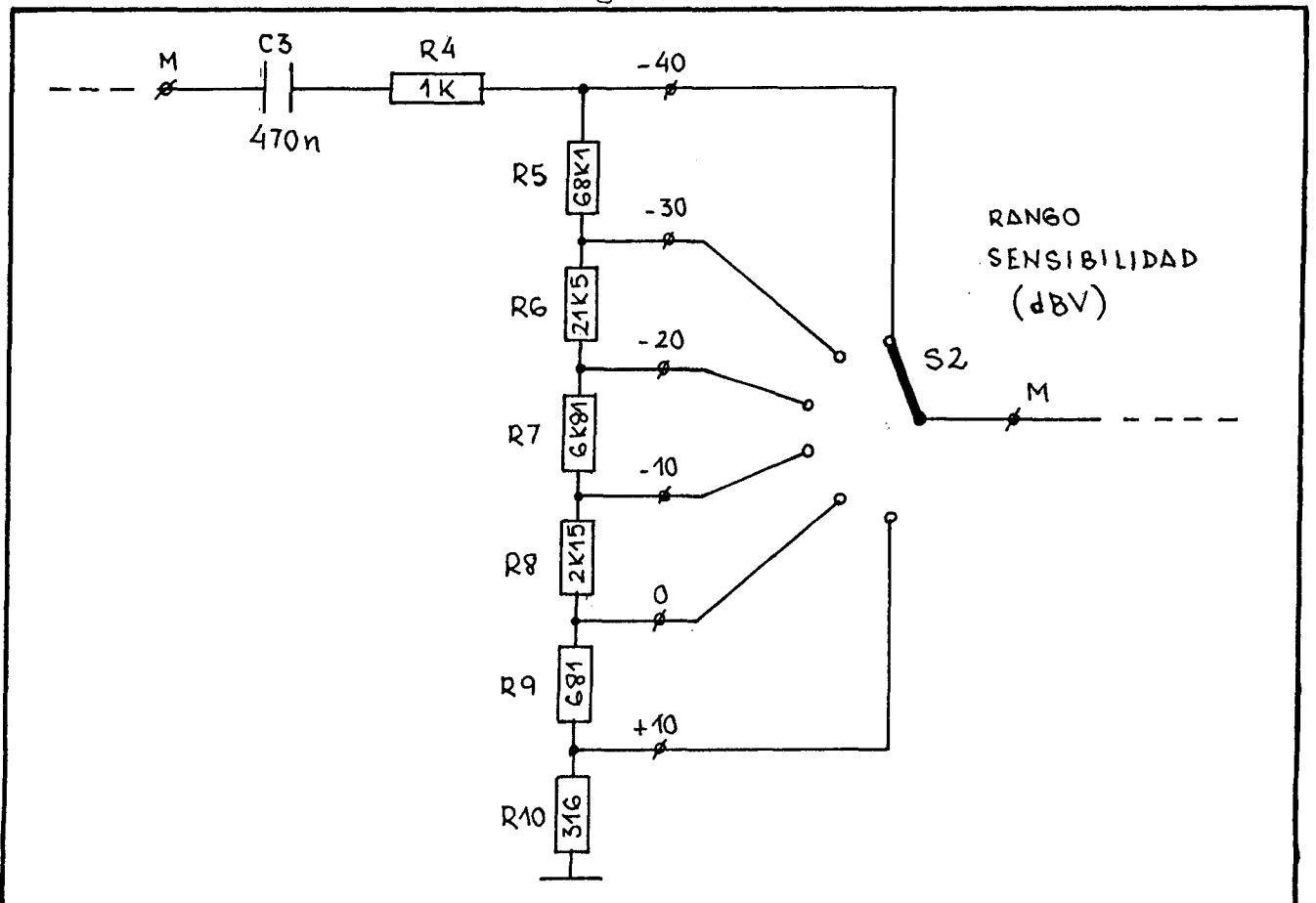


Fig. 5

#### 4.- CIRCUITO ATENUADOR.

El atenuador está constituido por el conmutador S2 y las resistencias R5....R10 (1%). Esta disposición se puede ver en la fig. (5).

Esta etapa está calibrada en unidades dBV (nivel absoluto de tensión):

$$\text{dBV} = 20 \cdot \log \frac{V}{V_0}$$

El OdBV corresponde a una tensión de 775 mv. eficaces.

Para esta etapa hemos elegido unos márgenes de atenuación de : -40, -30, -20, -10, 0 y +10 dBV.

A cada uno de estos márgenes le corresponde un valor de resistencia apropiado. Calcularemos a continuación estas resistencias fijando el valor de R10= 316 .

Para el margen de +10 dB :

$$-10 = 20 \cdot \log \frac{I(R10)}{I(R9+R10)} \quad (1)$$

donde I (R9+ R10)= 775mv. correspondiente al OdBV.

De la ecuación (1) despejamos el valor de R9::

$$R9 = \frac{R10(1 - 10^{-0.5})}{10^{-0.5}} \quad (2)$$

Sustituyendo el valor de R10 en la ecuación (2):

$$R9 = \frac{315(1 - 10^{-0.5})}{10^{-0.5}} = 681 \Omega$$

Para el margen de -10 dB:

$$10 = 20 \cdot \log \frac{I(R8 + R9 + R10)}{I(R9+R10)} \quad (3)$$

De la ecuación (3) despejamos el valor de R8:

$$R8 = (R9 + R10) \cdot (10^{0.5} - 1)$$

Sustituyendo el valor de R9 y R10:

$$R8 = (681 + 315) \cdot (10^{0.5} - 1) = 2153 \Omega .$$

Para el margen de -20dB:

$$20 = 20 \cdot \log \frac{I(R7 + R8 + R9 + R10)}{I \cdot (R9 + R10)} \quad (4)$$

Despejando de R7 de la ecuación (4):

$$R7 = (R9 + R10) \cdot (10 - 1) - R8$$

Sustituyendo los valores de R8, R9, y R10, nos queda:

$$R7 = 9 \cdot (681 + 315) - 2153 = 6811 \Omega .$$

Para el margen de -30 dB:

$$30 = 20 \log \frac{I(R6 + R7 + R8 + R9 + R10)}{I(R9 + R10)} \quad (5)$$

Despejando el valor de R6 de la ecuación (5) resulta:

$$R6 = (10^{3/2} - 1)(R9 + R10) - (R7 + R8)$$

Sustituyendo los valores de R7, R8, R9 y R10:

$$\begin{aligned} R6 &= (10^{3/2} - 1)(681 + 315) - (6811 + 2153) = \\ &= 21536 \Omega . \end{aligned}$$

Para el margen de -40dB:

$$40 = 20 \cdot \log \frac{I(R6 + R7 + R8 + R9 + R10 + R5)}{I(R9 + R10)}$$

Siguiendo el razonamiento anterior el valor de R5 será:

$$R5 = (10^2 - 1)(R9 + R10) - (R6 + R7 + R8)$$

Sustituyendo los valores de R6, R7, R8, R9 y R10, será:

$$R5 = (10^2 - 1) \cdot 996 - 30500 = 68104 \Omega .$$

Después de esta etapa atenuadora se encuentra un circuito amplificador que nos eleva el -

nivel de señal desde 7'75mv. eficaces (posición de -40 dBV) hasta unos 0'5v. eficaces que obtenemos a la salida.

Este bloque está compuesto por los amplificadores operacionales A2 y A3. La entrada del amplificador operacional A2 está protegida contra los picos de tensión por medio de la resistencia R4 (1 K) y los diodos D1 y D2 dispuestos en la configuración denominada diodos de sujeción, cuya misión es la de limitar la excursión de la salida de la etapa anterior a los valores comprendidos entre las tensiones de referencia impuestas por cada una de los diodos.

Presenta una resistencia de entrada -- R11, de elevado valor (10 M), a fin de poder traspasar en la mayor proporción posible los niveles de tensión de la etapa anterior. Esta etapa se caracteriza por una ganancia algo inferior a 6.

Observando la figura (6), calcularemos a continuación el valor de los componentes.

Al igual que antes:

$$V_o = A \cdot V_i \quad (1)$$

Del circuito se deducen las siguientes ecuaciones:

$$V_e = V_1 + V_i \quad (2)$$

$$V_o = V_2 + V_1 \quad (3)$$

Sustituyendo en la ecuación (2) la ecuación (1), tenemos:

$$V_e = \frac{V_o}{A} + V_1 \quad (4)$$

Debido a que A es muy grande en relación a  $V_o$ , se puede despreciar dicho término, con lo cual:

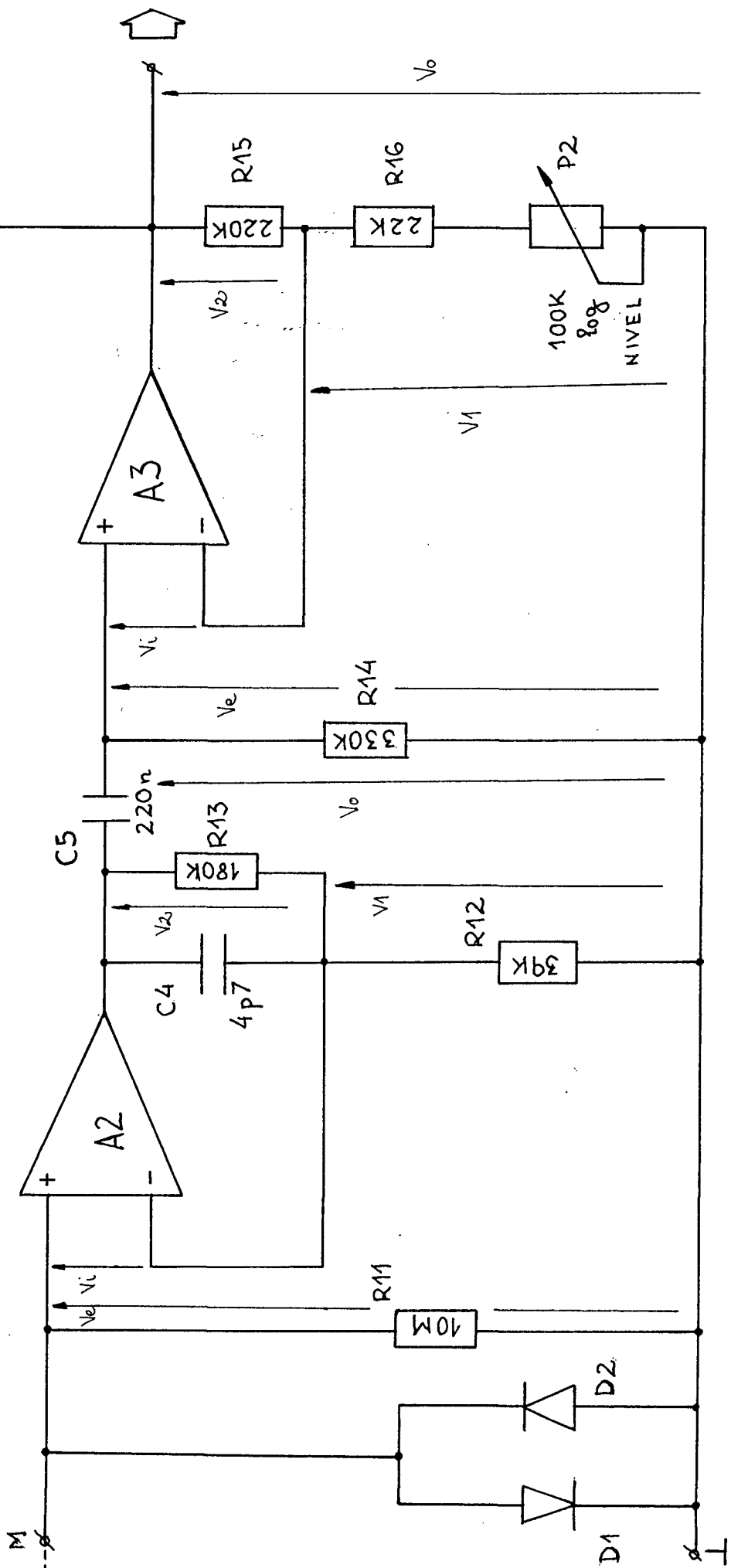


Fig. 6 y 7

$$V_e = V_1 \quad (5)$$

Sustituyendo en la ecuación (3) la (5)

nos queda:

$$V_o = V_2 + V_e \quad (6)$$

Como se observa:

$$V_2 = \frac{V_o}{R_{12} + R_{13}} \cdot R_{13} \quad (7)$$

Sustituyendo la ecuación (7) en la ecuación (6), resulta:

$$V_o = \frac{V_o}{R_{12} + R_{13}} \cdot R_{13} + V_e$$

de donde:

$$V_e = V_o \cdot \left(1 - \frac{R_{13}}{R_{12} + R_{13}}\right)$$

Con lo cual, la expresión de la ganancia queda:

$$A_v = V_o / V_e = \frac{R_{12} + R_{13}}{R_{12}} \quad (8)$$

Fijando el valor de  $R_{13} = 180 \text{ K}$  y  $A_v = 5.6$ , sustituyendo en la ecuación (8):

$$5.6 = \frac{180 \text{ K} + R_{12}}{R_{12}}$$

de donde:

$$(5.6 - 1) \cdot R_{12} = 180 \text{ K}$$

$$R_{12} = 39 \text{ K}$$

La red  $R_{13}/C_4$  tiene como función evitar las oscilaciones que puedan afectar al amplificador operacional.

El potenciómetro P2 permite ajustar la ganancia de la etapa siguiente (A3) entre 11 y 3.

Observando la fig.(7) deduciremos el valor de los componentes:

$$V_o = A \cdot V_i \quad (1)$$

$$V_e = V_i + V_2 \quad (2)$$

$$V_o = V_1 + V_2 \quad (3)$$

Sustituyendo la ecuación (1) en la (2)

convenientemente:

$V_e = \frac{V_o}{A} + V_2 = V_2$ , por ser  $V_o/A$  despreciable.

Del mismo modo:

$$V_2 = \frac{R_{16} + P_2}{R_{15} + R_{16} + P_2} \cdot V_o$$

de donde:

$$V_e = \frac{R_{16} + P_2}{R_{15} + R_{16} + P_2} \cdot V_o$$

con lo cual la ganancia queda:

$$A_v = \frac{V_o}{V_e} = \frac{R_{15} + R_{16} + P_2}{R_{16} + P_2}$$

Si  $P_2 = 0$ , lo que corresponde a  $A_v = 11$ , nos queda:

$$11 = \frac{R_{16} + R_{15}}{R_{16}}$$

Fijando  $R_{15} = 220 \text{ K}$ , y sustituyendo su valor:

$$11 = \frac{R_{16} + 220\text{K}}{R_{16}}$$

con lo que resulta el valor de  $R_{16} = 22\text{K}$ .

$$\text{Si } P_2 = 100\text{K}, \text{ tenemos } A_v = \frac{22 + 220 + 100}{22 + 100} = 3$$

La salida del amplificador operacional A3, alimenta directamente a los 30 filtros pasabanda.

## 5.- CIRCUITO DE PROTECCION.

Este circuito se puede ver en la figura(8). Su función es la de proteger contra las sobrecargas la salida de la etapa anterior. Cuando se produce este hecho se enciende el LED D4, indicando que hay sobremodulación en los amplificadores de entrada lo que se corrige reduciendo la señal de entrada o conmutando a una posición menos sensible.



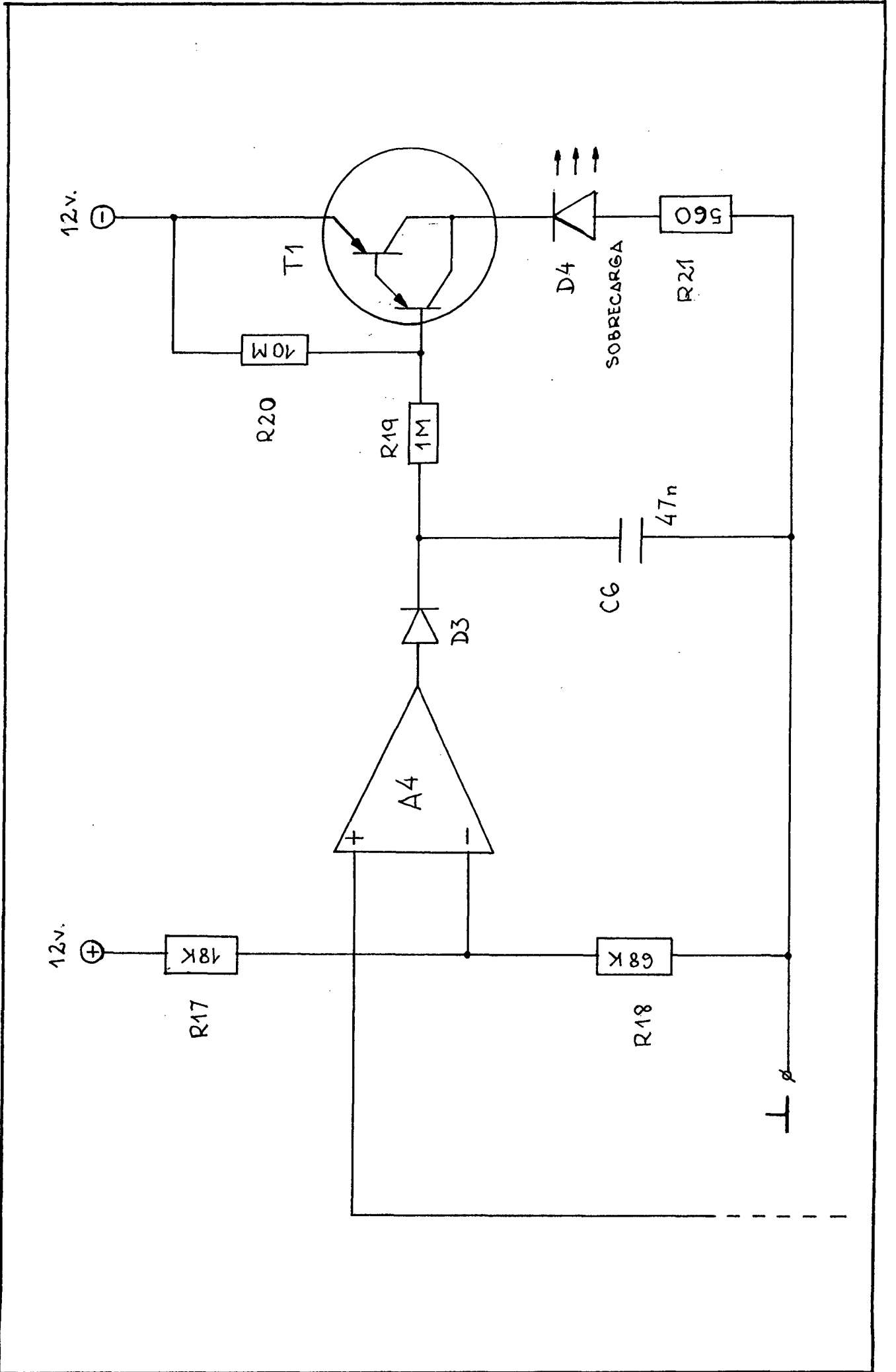


Fig. 8

El funcionamiento es el siguiente:

El -  
amplificador operacional A4 actúa de comparador, me-  
diante el divisor R17/R18, que proporciona la ten-  
sión de referencia necesaria para poder compararla  
con la salida de A3. La red D3/C6 actúa como circui-  
to de muestreo y bloqueo, alargando los picos de so-  
bremodulación para que el LED ilumine lo suficiente  
cuando los picos sean cortos.

El diodo D3 sólo deja pasar los nive-  
les de tensión positiva, rectificando la señal y --  
bloqueándose para los niveles de tensión negativa.

El condensador C6 se carga a través de  
las resistencias a la tensión de -12 voltios. Cuan-  
do la tensión de entrada supera la de referencia, -  
el amplificador operacional A4 dará a su salida a--  
proximadamente 12 voltios, tendiendo a cargar con -  
una constante de tiempo muy pequeña el condensador  
C6 a +12 voltios, con lo cual se produce la satura-  
ción del Darlington con una corriente de base peque-  
ña, debido a que tiene una hfe de 30K. Al saturar  
el transistor, éste activa el LED e indica la sobre-  
modulación.

Si la tensión de entrada disminuye por  
debajo de la de referencia, el diodo no conduce pe-  
ro el condensador mantiene saturado al transistor.

Para calcular los valores de los compo-  
nentes, fijaremos la tensión de referencia a  $V_r=9.5$   
voltios y  $R_{17}=18K$ , con lo cual:

$$I = \frac{12 - 9.5}{18K} = 0.138mA.$$

Como:

$$R18 = V_r / I = 9'5 / 0'138 = 68K$$

La R21 limita la corriente de colector del Darlington. Si tenemos en cuenta, que la tensión que cae en el diodo es de 2'3 voltios y la tensión colector-emisor de saturación es de 0'2 voltios, poniendo R21= 560 , la intensidad de saturación será de:

$$I_{Csat.} = \frac{12 - 2'3 - 0'2}{560} = 17mA.$$

La corriente de base mínima para saturar al transistor será:

$$I_{Bsat.} = \frac{I_{Csat.}}{hfe} = \frac{17 \cdot 10^{-3}}{30 \cdot 10^3} = 0'56 A.$$

Las R19 y R20 limita la primera y deriva la segunda, parte de la corriente de base; a la vez que forman parte de la constante de tiempo de carga a -12 voltios. Por ello sus valores son:

$$R19 = 1M, R20 = 10M$$

## 6.- CALCULO DE LOS CONDENSADORES DE ACOPLO.

El uso de estos condensadores es fundamental si queremos obtener un buen nivel de señal a la salida de la etapa. Su misión consiste en cortocircuitar la componente continua de la tensión y facilitar al máximo el paso de la componente alterna.

En este caso lo que nos interesa son niveles de tensión y no de potencia, por lo que calcularemos estos condensadores teniendo en cuenta -- que su impedancia sea 10 veces inferior a la suma de las impedancias de entrada y salida de cada amplificador, a la frecuencia de 20 Hz. De esta forma:

$$C1 = (2 \cdot \pi \cdot 20 \cdot 9'4 \cdot 10^3)^{-1}$$

$$C1 = 846,56 \text{ nF.}$$

$$C3 = (2 \cdot \pi \cdot 20 \cdot 10^6)^{-1} = 8 \text{ nF.}$$

$$C5 = (2 \cdot \pi \cdot 20 \cdot 33 \cdot 10^3)^{-1} = 240 \text{ nF}$$

## 7.- LOS FILTROS.

### 7.1.- FILTRO PASABANDA RESONANTE ACTIVO.

Se denomina filtro pasabanda a un circuito electrónico que permite el paso a su través de todas las señales cuyas frecuencias estén comprendidas entre dos valores determinados, llamados frecuentemente frecuencias de corte del filtro.

Con un filtro pasabanda puede eliminarse señales no útiles, espúreas ó indeseables que vayan mezcladas con la señal que se desea, de forma que esta última pueda recuperarse de entre el conjunto con la condición de que sean de frecuencia distinta de las no deseadas.

Los parámetros más importantes de un filtro pasabanda son la ganancia en la banda de paso, la anchura de la banda de paso (determinada por las frecuencias de corte) y la pendiente de filtro o grado de atenuación que experimentan las señales fuera de la banda de paso.

La ganancia suele expresarse como número abstracto o en desibelios, la anchura de banda en hertzios y la pendiente en desibelios por octava. Se suele usar el factor Q que es un concepto en el que se engloba la pendiente y la anchura de banda a un tiempo. Esto se puede ver en las figuras (9) y (10).

La separación de las distintas señales

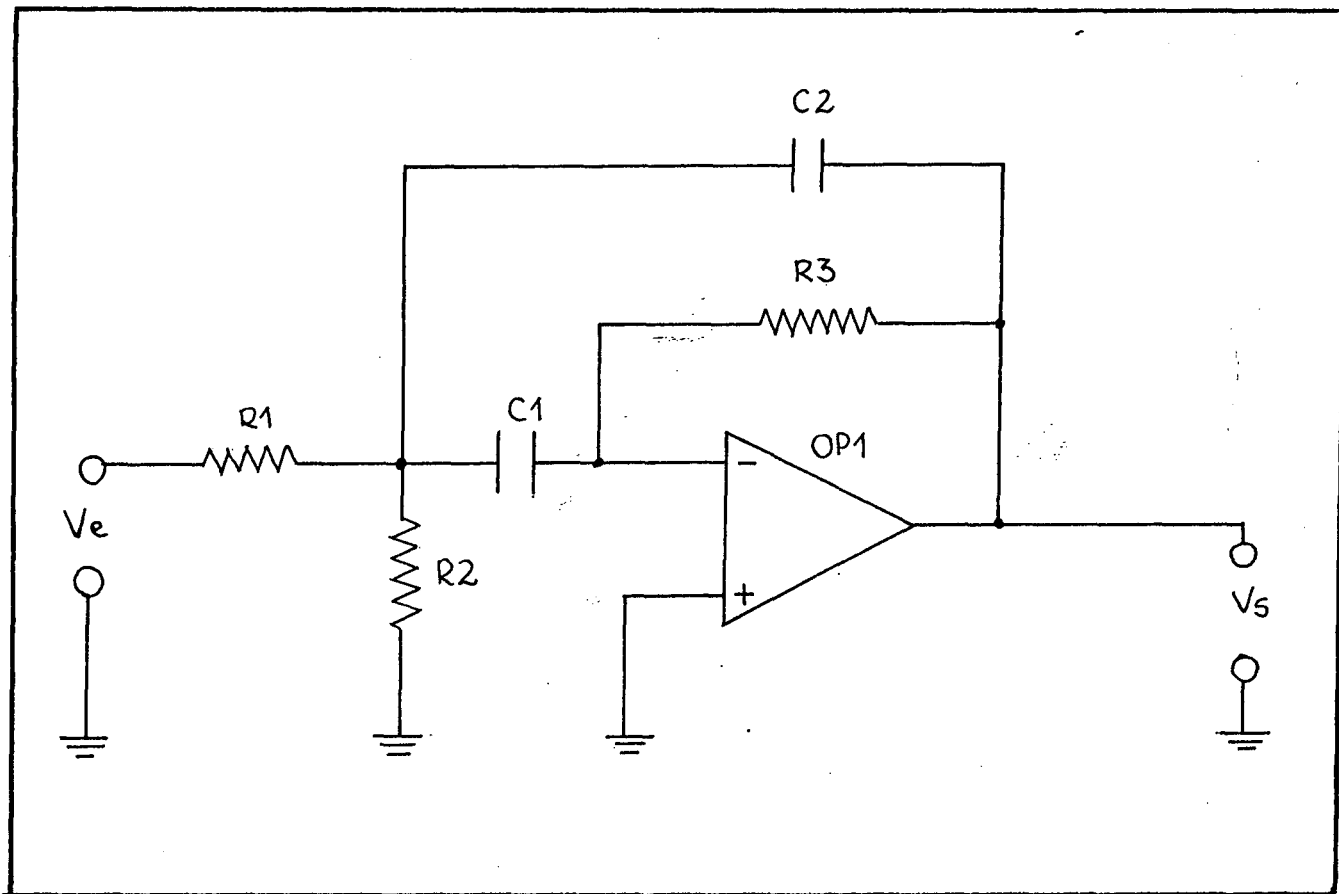


Fig. 9

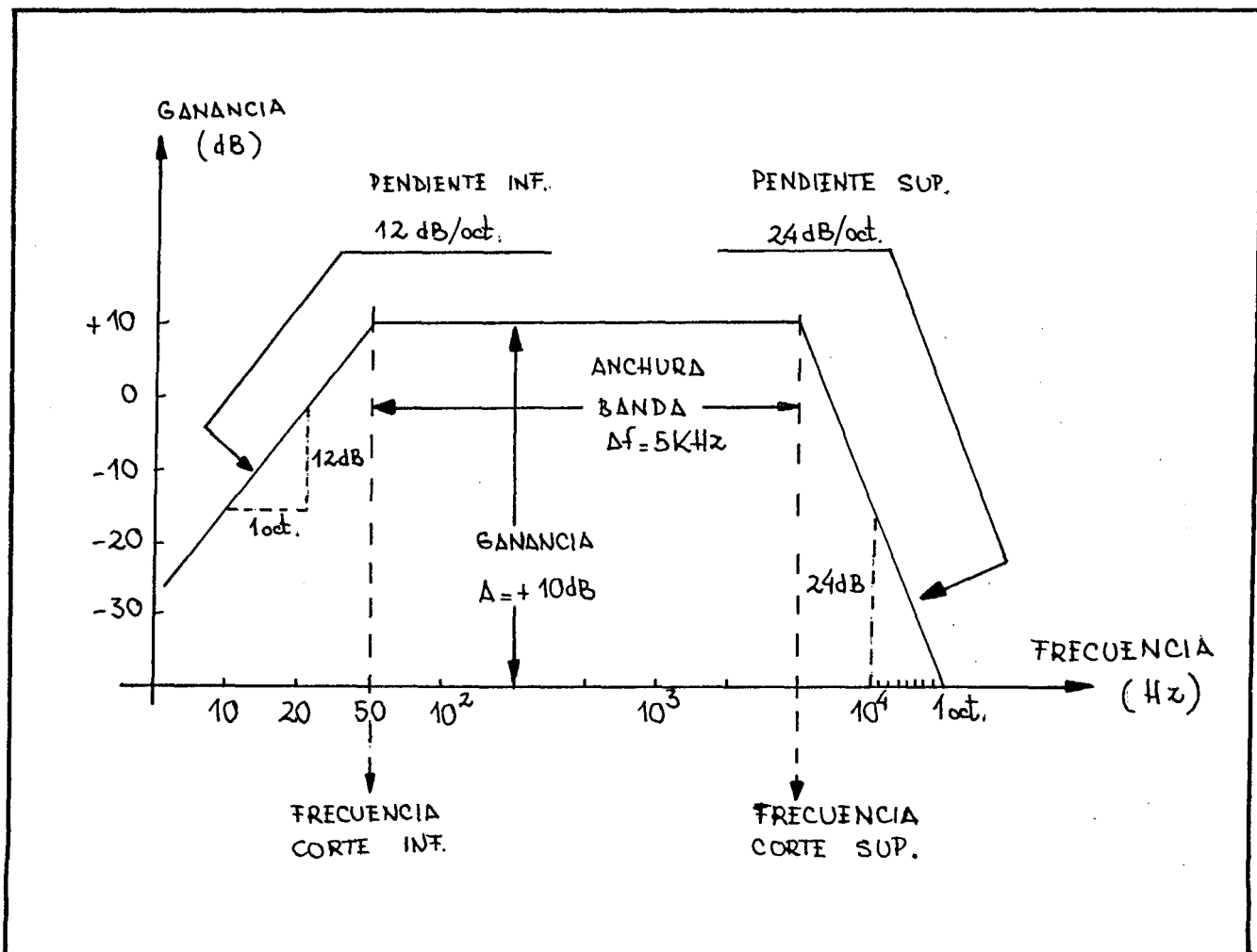


Fig. 10

del espectro de audio puede llevarse a cabo mediante el empleo de componentes activos como transistores, amplificadores operacionales ( tal es nuestro caso) y otros. La combinación de ciertos elementos puede lograr una separación mucho más exacta y precisa que la lograda con componentes pasivos solamente.

### 7.2.- DESCRIPCIÓN TEORICA.

Una descripción general del conjunto - sin entrar en estudios matemáticos nos muestra - que el circuito es una especie de combinación de un filtro pasa-alto y otra pasa-bajo. Para frecuencias de señal muy altas el condensador C1 presenta una débil impedancia, haciendo que C2 y R3 queden en paralelo, y el circuito actúa como un filtro-bajo. Para frecuencias muy bajas, la impedancia del condensador C2 es muy elevada y no tiene efecto en el circuito haciendo que C1, R3 y el operacional actúen como un filtro pasa-alto.

Existirá una gama de frecuencias intermedias en la que el conjunto actúa como amplificador habiéndose eliminado todas las señales de frecuencias altas y bajas. El circuito se comporta, por tanto, como un filtro pasabanda.

Colocando dos de ellos, uno a continuación del otro (en cascada) pueden construirse -- filtros muy selectivos.

### 7.3.- DEMOSTRACION MATEMATICA.

Se obtiene un prototipo de filtro pasabanda de segundo orden, colocando en cascada un

filtro de paso bajo de segundo orden con frecuencia de corte  $f_{oH}$  y un filtro de paso alto de segundo orden con frecuencia de corte  $f_{oL}$ , y teniendo en cuenta que  $f_{oH} > f_{oL}$ , tal como se indica en la fig. (11).

Se observa que este filtro tiene respuesta constante para  $f_{oL} < f < f_{oH}$  y ganancia nula fuera de esta banda. Normalmente, el filtro pasabanda tiene una respuesta, cuyo pico se haya en una frecuencia central ( $f_o$ ) y desciende a ambos lados de  $f_o$ . Cabe obtener una aproximación sencilla de la característica de banda estrecha empleando un circuito resonante LC. Un prototipo fundamental de filtro resonante de segundo orden, cuya función de transferencia se va a deducir es la siguiente: Si suponemos que el amplificador tiene una ganancia de  $A_o = V_o/V_i$  positiva y constante para todas las frecuencias, tenemos:

$$A_v(j\omega) = \frac{V_o}{V_s} = \frac{V_o \cdot V_i}{V_s \cdot V_i} = \frac{R \cdot A_o}{R + j\omega(L - 1/\omega C)} \quad (1)$$

La frecuencia central o de resonancia  $f_o = \omega_o/2\pi$ , se define como la frecuencia a la cual la inductancia resuena con la capacidad, o sea aquella a la que las reactancias inductiva y capacitiva son iguales (en magnitud), es decir:

$$\omega_o^2 = \frac{1}{LC} \quad (2)$$

Conviene definir el factor de calidad  $Q$  de este circuito:

$$Q = \frac{\omega_o \cdot L}{R} = (\omega_o \cdot C \cdot R)^{-1} = \frac{1}{R} (L/C)^{1/2} \quad (3)$$

Sustituyendo la ecuación (3) en la (1), obtenemos el valor y la fase de la función -

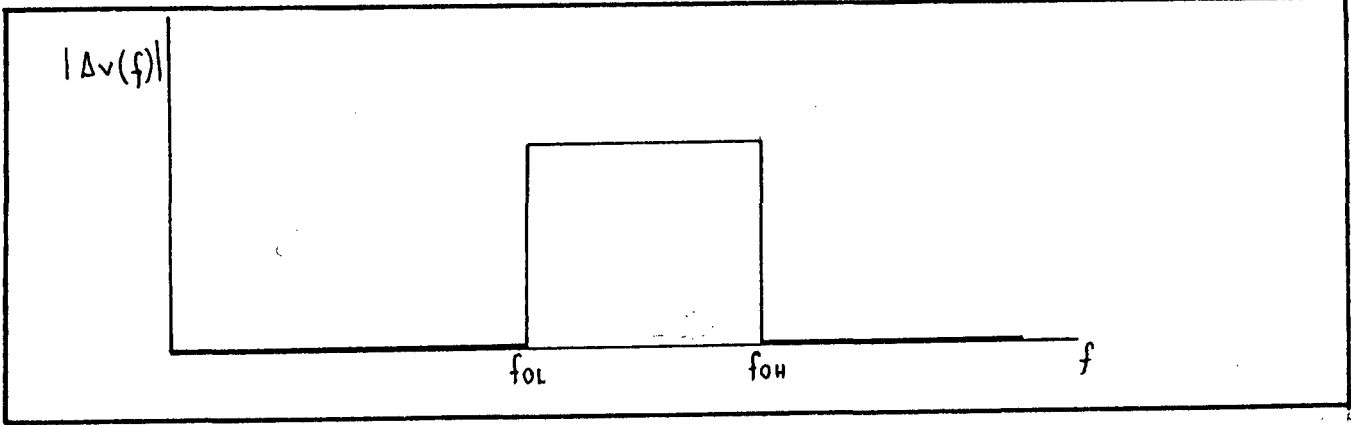


Fig. 11

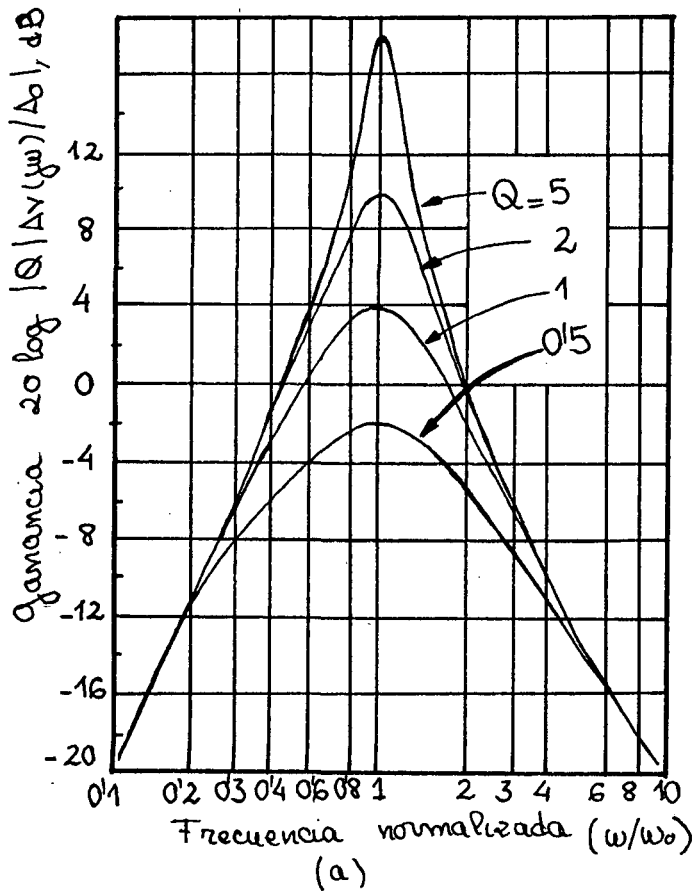
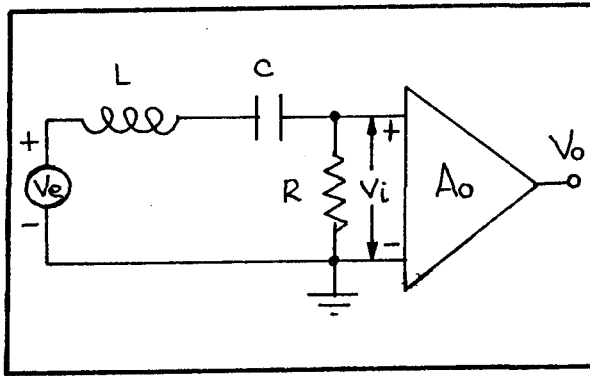


Fig. 12

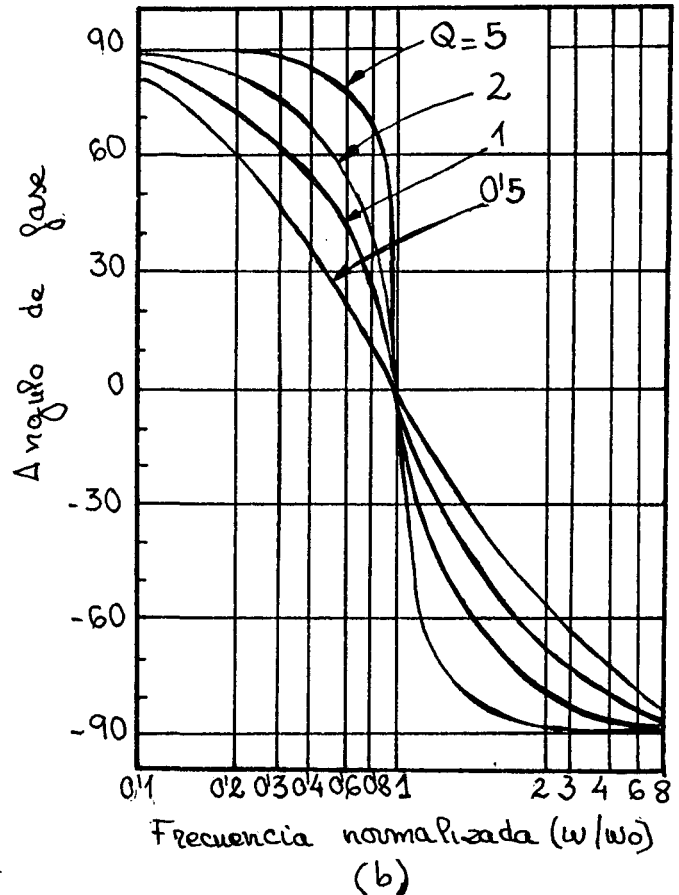


Fig. 13



de transferencia:

$$|Av(jw)| = \frac{A_0}{\left[ 1 + Q^2 \left( \frac{w}{w_0} - \frac{w_0}{w} \right)^2 \right]^{1/2}}$$

$$\theta(w) = - \arctg Q \left( \frac{w}{w_0} - \frac{w_0}{w} \right)$$

Las curvas correspondientes a estas fórmulas son las fig. (12) y (13), respectivamente.

- Simetría Geométrica:

En las curvas  $|Av(jw)|$  de la ganancia se observa que para cualquier frecuencia  $w' < w_0$ , existe otra frecuencia  $w'' > w_0$  a la que  $|Av(jw)|$  tiene el mismo valor. Demostraremos que la media geométrica de estas frecuencias es  $w_0$ , es decir que:

$$w_0^2 = w' \cdot w''$$

$$\text{Haciendo } |Av(jw')| = |Av(jw'')|$$

obtendremos:

$$\frac{w'}{w_0} - \frac{w_0}{w'} = - \left( \frac{w''}{w_0} - \frac{w_0}{w''} \right), \quad (4)$$

El signo menos se coloca fuera del paréntesis debido a que:

$$w' < w_0 < w''$$

De la ecuación (4) deducimos:  $w_0^2 = w' \cdot w''$  del siguiente modo:

$$w'^2 \cdot w'' - w_0^2 \cdot w'' = w_0^2 \cdot w' - w''^2 \cdot w'$$

$$w'^2 \cdot w'' + w''^2 \cdot w' = w_0^2 \cdot (w'' + w')$$

$$(w' \cdot w'' + w''^2) \cdot w' = w_0^2 \cdot (w'' + w')$$

$$(w' + w'') \cdot w' \cdot w'' = w_0^2 \cdot (w'' + w')$$

$$w_0^2 = w' \cdot w'' \quad (5)$$

- Ancho de banda:

Sean  $w_1 < w_0$  y  $w_2 > w_0$  las dos frecuencias a ambos lados de  $w_0$ , para los que la caída de ganancia es de 3 dB a partir de su valor  $A_0$  a  $w_0$ . Se define entonces el ancho de banda como:

$$B = \frac{w_2 - w_1}{2\pi} = \frac{1}{2\pi} \left( w_2 - \frac{w_0^2}{w_2} \right) \quad (6).$$

$$w_0^2 = w_1 \cdot w_2$$

La frecuencia  $w_2$  se halla haciendo:  $\left| \frac{A_v(jw)}{A_0} \right| = \frac{1}{\sqrt{2}}$  (7)

De la ecuación (4) se deduce que:

$$\begin{aligned} Q \left( \frac{w_2}{w_0} - \frac{w_0}{w_2} \right) &= 1 = \\ &= \frac{Q}{w_0} \left( w_2 - \frac{w_0^2}{w_2} \right) \end{aligned} \quad (8)$$

Comparando la ecuación (6) con la ecuación (8) tenemos que:

$$B = \frac{1}{2\pi} \cdot \frac{w_0 R}{w_0 L} = \frac{1}{2\pi} \cdot \frac{R}{L}$$

- Filtro pasabanda activo RC:

La forma general de un -- filtro de pasabanda de segundo orden se obtiene haciendo  $s=jw$  en la ecuación (1) con lo cual queda:

$$\begin{aligned} A_v(s) &= \frac{R \cdot A_0}{R + sL + 1/sC} = \\ &= \frac{(R/L) \cdot A_0 \cdot s}{s^2 + s(R/L) + 1/LC} \end{aligned} \quad (9)$$

Sabiendo que:

$$\omega_o^2 = \frac{1}{LC}$$

$$Q = \frac{\omega_o L}{R} = (\omega_o C R)^{-1} = 1/R \cdot (L/C)^{1/2}$$

Sustituyendo estas dos ecuaciones en la ecuación (9):

$$Av(s) = \frac{(\omega_o/Q) \cdot A_o \cdot s}{s^2 + s \cdot \frac{\omega_o}{Q} + \omega_o^2} \quad (10)$$

Esta función de transferencia obtenida del circuito RLC puede obtenerse con el circuito de realimentación múltiple en el que se emplean dos condensadores, tres resistencias y un amplificador operacional sin inductancias, fig. (14).

Debido a la tierra virtual en la entrada del amplificador operacional la tensión a través de R3 es Vo y la corriente I3 es Vo/R3. Si suponemos que la corriente de polarización es despreciable, V' vendrá dada por:

$$V' = - \frac{I3}{sC} = - \frac{V_o}{s \cdot C \cdot R3} \quad (11)$$

Aplicando la ley de Kirchhoff al nudo V':

$$V' = V_e - I \cdot R1 = - \frac{V_o}{s \cdot C \cdot R3} \quad (12)$$

$$I = I1 - I2 - I3 \quad (13)$$

$$I = V'/R2 \quad (14)$$

$$I = V_e - V'/R1 \quad (15)$$

$$I2 = (V_o - V') \cdot s \cdot C \quad (16)$$

Sustituyendo las ecuaciones (14), (15), e I3 en la ecuación (13) nos queda:

$$I = \frac{V'}{R2} - (Vo - V').s.C - \frac{Vo}{R3}, \quad (17)$$

Sustituyendo la ecuación (11) en la (17):

$$I = - \frac{Vo}{s.C.R3.R2} - (Vo + \frac{Vo}{s.C.R3}).s.C - \frac{Vo}{R3} \quad (18)$$

Quitando el paréntesis -- queda:

$$I = - \frac{Vo}{s.C.R3.R2} - Vo.s.C - \frac{Vo}{R3} - \frac{Vo}{R3}, \quad (19)$$

$$I = - \frac{Vo}{s.C.R3.R2} - Vo.s.C - 2 \frac{Vo}{R3} \quad (20)$$

$$I = - Vo \left( \frac{1}{s.C.R3.R2} + s.C + \frac{2}{R3} \right), \quad (21)$$

Sustituyendo la ecuación (15) en la (21):

$$\frac{Ve - V'}{R1} = - Vo \left( \frac{1}{s.C.R3.R2} + s.C + \frac{2}{R3} \right) \quad (22)$$

Reagrupando convenientemente los términos:

$$\frac{Vo}{s.C.R3} + Ve = - Vo.R1 \left( \frac{1}{s.C.R2.R3} + s.C + \frac{2}{R3} \right), \quad (23)$$

$$Ve = - Vo \left( \frac{R1}{s.C.R2.R3} + s.C.R1 + 2 \frac{R1}{R3} + (s.C.R3)^{-1} \right), \quad (24)$$

$$Ve = - Vo \left( \frac{R1 + R2 + 2.R1.R2.s.C + R1}{s.C.R2.R3} \right)$$

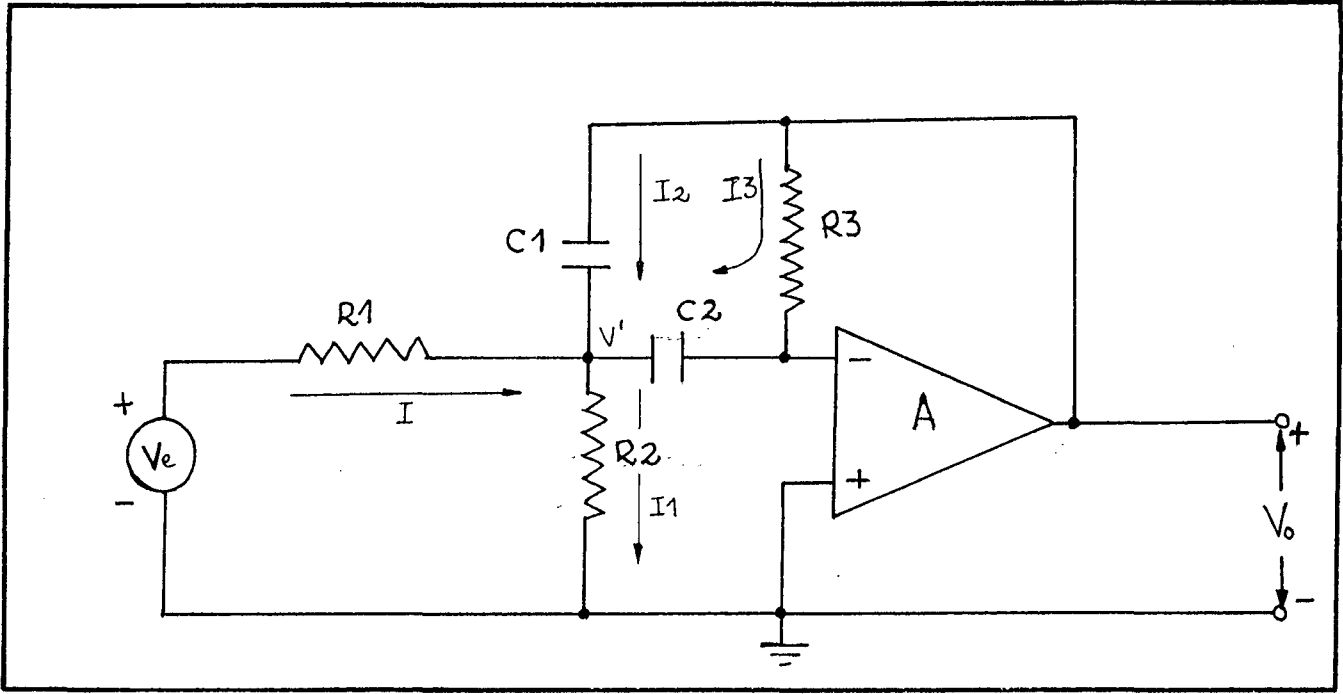


Fig. 14

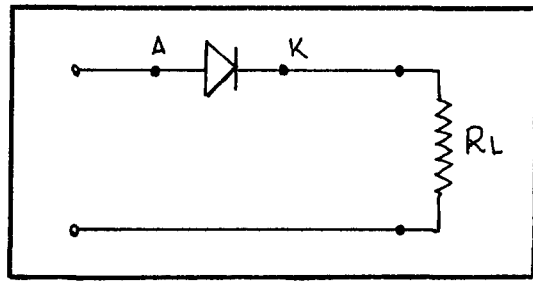


Fig. 15

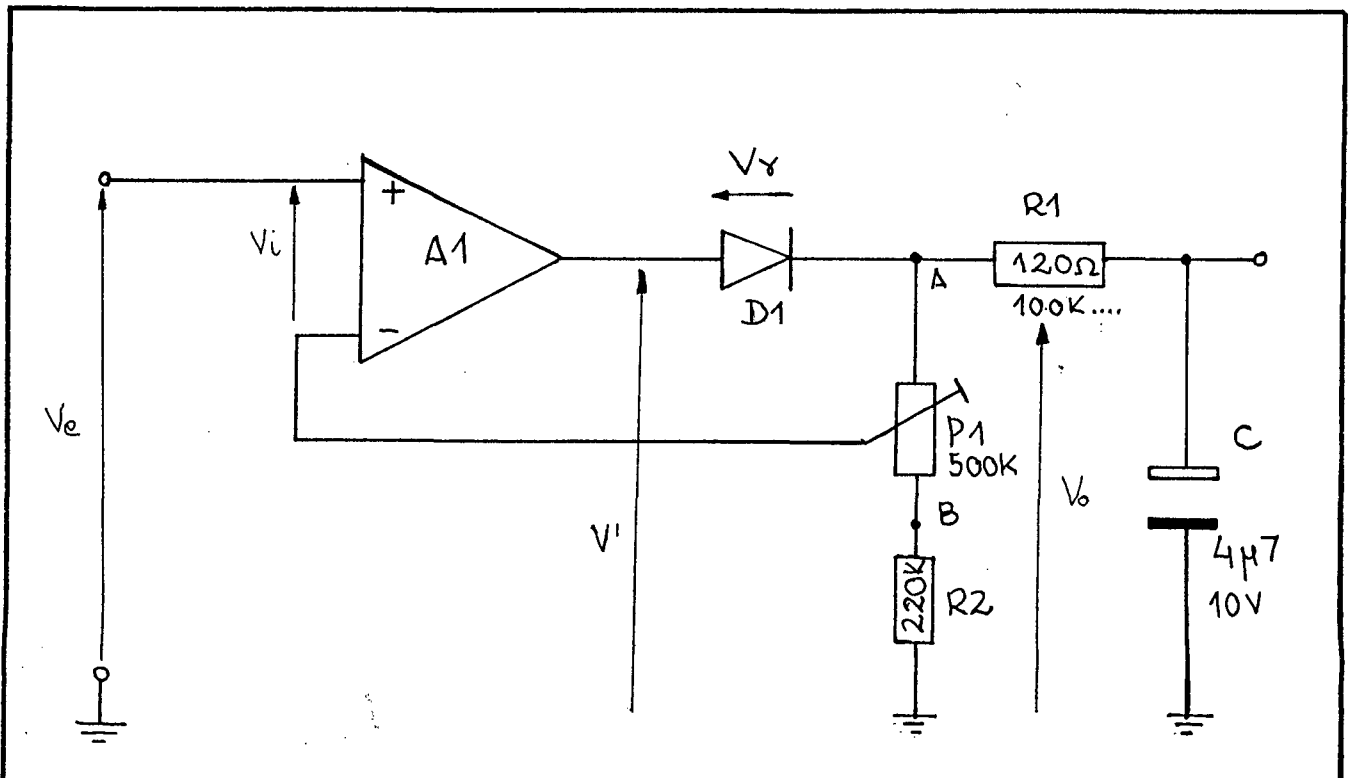


Fig. 16

$$\frac{.R2.R3.s^2.C^2}{(25)}$$

Multiplicando por  $(C^2.R1.R2.R3)$ :

$$Ve = - Vo \left( \frac{s^2 + 2s/C.R3 + (C^2.R2.R3)^{-1} + (C^2.R1.R3)^{-1}}{s/C.R1} \right) \quad (26)$$

Siendo  $R' = R1.R2/R1+R2$ ,

nos queda:

$$Ve = - Vo \left( \frac{s^2 + 2s/C.R3 + (C^2.R3)^{-1}.1/R'}{s/C.R1} \right) \quad (27)$$

Con lo cual nos queda:

$$Av = \frac{Vo(s)}{Ve(s)} = - \frac{s/C.R1}{s^2 + (2/R3.C).s + 1/R'}. \quad (28)$$

#### FILTRO PASABANDA DE 2º ORDEN

#### 7.4.- DEDUCCION DE LAS FORMULAS PARA EL CALCULO DE LOS COMPONENTES.

Se puede observar que la ecuación -- (28) tiene un cero en el origen, además de dos polos. Igualando los coeficientes de s en los numeradores de las ecuaciones (10) y (28), tenemos:

$$- (1/C.R1) = (w_0/Q).Ao , \text{ de donde:}$$

$$C.R1 = \frac{Q}{w_0 (-Ao)} \quad (29)$$

Igualando los coeficientes de s en los denominadores de esas mismas ecuaciones:

$$\frac{R_3 \cdot C}{2} = \frac{Q}{w_0} \quad (30)$$

Iguando los términos constantes - en los denominadores de dichas ecuaciones, se obtiene:

$$R' \cdot R_3 \cdot C^2 = \frac{1}{w_0^2} \quad (31)$$

Dividiendo esta última ecuación por la fórmula  $(R_3 \cdot C)/2 = Q/w_0$ , se elimina  $R_3$ :

$$\frac{R' \cdot R_3 \cdot C^2}{\frac{R_3 \cdot C}{2}} = \frac{1}{\frac{w_0^2}{Q/w_0}}$$

$$2 \cdot R' \cdot C = 1/Q \cdot w_0$$

Con lo cual la ecuación quedará:

$$2 \cdot R' \cdot C = \frac{1}{Q \cdot w_0} \quad (32)$$

Cualesquiera valores reales y positivos de  $R_1$ ,  $R'$ ,  $R_3$ ,  $C_1$  y  $C_2$  que satisfagan las -- ecuaciones (29), (30) y (32) se pueden aceptar para diseñar un filtro pasabanda activo. Como sólo -- tenemos tres ecuaciones para cinco parámetros, dos de ellos se pueden elegir arbitrariamente (por ejemplo  $C_1$  y  $C_2$ ) que en nuestro caso tienen el mismo valor  $C$ .

#### 7.5.- DEDUCCION DE LAS FORMULAS PARA EL CALCULO DE LAS FRECUENCIAS.

Cada uno de los tres amplificadores operacionales de un filtro se presenta como un filtro pasabanda de realimentación múltiple.

Sus frecuencias centrales están desplazadas entre sí, de modo que si la frecuencia -- central de uno de los filtros corresponde a la frecuencia central de 1/3 de octava los de los otros dos estarán exactamente en los puntos de corte de esta banda. Un cálculo del factor de calidad Q y - de la ganancia de cada filtro permite linealizar - la parte superior de la curva obtenida.

La ganancia de los filtros centrales (los que corresponden a  $f_0$ ) son unitarias mientras que la de los laterales que corresponden a las frecuencias de corte son de  $1/(2)^{1/2}$ .

Como el filtro lateral superior de -- una banda dada es idéntico al filtro lateral inferior de la banda superior, se han puesto filtros - laterales comunes a filtros de 1/3 de octava, ahorrando así 15 filtros laterales.

Los filtros de 1/3 de octava siempre cumplen la relación siguiente:

$$\frac{f_2}{f_1} = 2^{1/3} \quad (1)$$

siendo  $f_1$  y  $f_2$  las frecuencias de umbral de -3dB.

Llamando  $f_0$  a la frecuencia central, tenemos:

$$f_0 = (f_1 \cdot f_2)^{1/2} \quad (2)$$

De la ecuación (1) se deduce:

$$f_2 = 2^{1/3} \cdot f_1 \quad (3)$$

De la ecuación (2) se deduce:



$$f_0^2 = f_1 \cdot f_2 \quad (4)$$

Sustituyendo la ecuación (3) en la ecuación (4):

$$f_0^2 = f_1^2 \cdot 2^{1/3}$$

$$f_1^2 = f_0^2 \cdot 2^{-1/3}$$

$$f_1 = f_0 \cdot 2^{-1/6}$$

$$f_2 = f_0 \cdot 2^{1/6}$$

La banda de frecuencias está definida por:

$$f_0 = 10^{n/10} \text{ Hz.}$$

donde  $n = n^0$  de la banda = 14, 15, ..... , 43.

Matemáticamente se demuestra que:

$$2 = 10^{\log 2} = 10^{0.3}$$

$$2 = 10^{3/10}$$

$$2^{\pm 1/6} = 10^{3/10 \cdot \pm 1/6} = 10^{\mp 1/20}$$

Aplicando este cálculo a las ecuaciones de  $f_1$  y  $f_2$ :

$$f_1 = f_0 \cdot 2^{-1/6} = f_0 \cdot 10^{-1/20} = 10^{2n/20} \cdot 10^{-1/20}$$

Con lo cual nos queda:

$$f_1 = 10^{(2n - 1)/20}$$

$$f_2 = 10^{(2n + 1)/20}$$

#### 7.6.- CALCULO DE LOS COMPONENTES DE CADA FILTRO.

Calcularemos primero la frecuencia central y las dos adyacentes correspondientes a cada uno de los filtros. Para ello aplicando las fórm

mulas deducidas con anterioridad, es decir:

$$f_0 = 10^{n/10} \text{ Hz. donde } n = 14, \dots, 43$$

$$f_1 = 10^{(2n - 1)/20} \text{ Hz.}$$

$$f_2 = 10^{(2n + 1)/20} \text{ Hz.}$$

calculamos la tabla siguiente:

valor n <sup>o</sup>	frec. central	frec. sup.	f.inf.
n	f <sub>0</sub> (Hz)	f <sub>2</sub> (Hz)	f <sub>1</sub> (Hz)
14	25	28	22
15	32	36	28
16	40	44	36
17	50	56	44
18	63	70	56
19	80	89	70
20	100	112	89
21	125	141	112
22	160	178	141
23	200	224	178
24	250	282	224
25	315	355	282
26	400	446	355
27	500	562	446
28	630	708	562
29	800	891	708
30	1K	1122	891
31	1K25	1412	1122
32	1K6	1778	1412
33	2K	2238	1778
34	2K5	2818	2238
35	3K15	3548	2818

36	4K	4467	3548
37	5K	5623	4467
38	6K3	7079	5623
39	8K	8912	7079
40	10K	11220	8912
41	12K5	14125	11220
42	16K	17783	14125
43	20K	22387	17783

Una vez halladas todas las frecuencias pasaremos a calcular los valores de las resistencias y los condensadores, empleando las ecuaciones que deducimos en el otro apartado. Estas ecuaciones son:

$$R1 = \frac{Q}{\omega_o \cdot A \cdot C}$$

$$R' = ( 2 \cdot \omega_o \cdot C \cdot Q )$$

$$R3 = \frac{2 \cdot Q}{\omega_o \cdot C}$$

$$R2 = \frac{R' \cdot R1}{R1 - R'}$$

Los valores de los condensadores se fijan a conveniencia para cada uno de los filtros. El factor Q de todos los filtros es algo superior a 4, mientras que las ganancias de los filtros "centrales" ( $f_o$ ) son unitarias y la de los filtros "laterales" ( $f1$  y  $f2$ ) de 1'4.

Desde el punto de vista teórico hubiera sido mejor obtener una pendiente todavía mayor para aumentar la precisión, pero esto requiere un aumento del factor Q que no se puede obtener -

con los amplificadores operacionales de bajo coste que hemos utilizado.

Una vez dicho todo esto, pasemos a calcular la tabla de valores siguiente que engloba todos los componentes que forman los 30 filtros:

	PLACA I	PLACA II	PLACA III	PLACA IV
f(Hz)	28	178	1122	7079
Q	4'16	4'16	4'16	4'16
A	1'4	1'4	1'4	1'4
R1	76K8	26K7	42K2	66K5
R2	3K24	1K13	1K78	2K8
R3	215K	75K	118K	187K
C1,C2	220n	100n	10n	1n
f(Hz)	22	141	891	5623
Q	4'16	4'16	4'16	4'16
A	1'4	1'4	1'4	1'4
R4	95K3	34K	53K6	84K5
R5	4K02	1K43	2K26	3K57
R6	267K	93K1	150K	237K
C3,C4	220n	100n	10n	1n
f(Hz)	25	160	1K	6K3
Q	4'16	4'16	4'16	4'16
A	1	1	1	1
R7	118K	41K2	66K5	53K6
R8	3K57	1K27	2K	2K21
R9	237K	82K5	130K	147K
C5,C6	220n	100n	10n	1n
f(Hz)	36	224	1412	8912

Q	4'16	4'16	4'16	4'16
A	1'4	1'4	1'4	1'4
R10	60K4	21K	33K2	82K5
R11	2K55	887 $\Omega$	1K4	2K21
R12	169K	59K	93K1	147K
C7,C8	220n	100n	10n	1n
f(Hz)	32	200	1K25	8K
Q	4'16	4'16	4'16	4'16
A	1	1	1	1
R13	95K3	32K7	52K3	82K5
R14	2K87	1K	1K58	2K49
R15	187K	66K5	105K	165K
C9,C10	220n	100n	10n	1n
f(Hz)	44	282	1778	11220
Q	4'16	4'16	4'16	4'16
A	1'4	1'4	1'4	1'4
R16	48K7	76K8	26K7	42K2
R17	2K05	3K24	1K13	1K78
R18	133K	215K	25K	118K
C11,C12	220n	22n	10n	1n
f(Hz)	36	224	1412	8912
Q	4'16	4'16	4'16	4'16
A	1'4	1'4	1'4	1'4
R19	60K4	95K3	34K	53K6
R20	2K55	4K02	1K43	2K26
R21	169K	267K	93K1	150K
C13,C14	220n	22n	10n	1n

f(Hz)	40	250	1K6	10K
Q	4'16	4'16	4'16	4'16
A	1	1	1	1
R22	75K	118K	41K2	66K5
R23	2K26	3K57	1K27	2K
R24	150K	237K	82K5	130K
C15,C16	220n	22n	10n	1n

f(Hz)	56	355	2238	14125
Q	4'16	4'16	4'16	4'16
A	1'4	1'4	1'4	1'4
R25	38K3	60K4	21K	33K2
R26	1K62	2K55	887Ω	1K4
R27	107K	169K	69K	93K1
C17,C18	220n	22n	10n	1n

f(Hz)	50	315	2K	12K5
Q	4'16	4'16	4'16	4'16
A	1	1	1	1
R28	59K	75K	76K8	52K3
R29	1K82	2K26	3K24	1K58
R30	118K	150K	215K	105K
C19,C20	220n	22n	2n2	1n

f(Hz)	70	446	2818	17783
Q	4'16	4'16	4'16	4'16
A	1'4	1'4	1'4	1'4
R31	66K5	48K7	76K8	26K7
R32	2K8	2K05	3K24	1K13
R33	187K	133K	215K	75K
C21,C22	100n	22n	2n2	1n

f(Hz)	56	355	2238	14125
Q	4'16	4'16	4'16	4'16
A	1'4	1'4	1'4	1'4
R34	84K5	60K4	95K3	34K
R35	3K57	2K55	4K02	1K43
R36	237K	169K	267K	93K1
C23,C24	100n	22n	2n2	1n
f(Hz)	63	400	2K5	16K
Q	4'16	4'16	4'16	4'16
A	1	1	1	1
R37	105K	75K	118K	41K2
R38	3K16	2K26	3K57	1K27
R39	210K	150K	237K	82K5
C25,C26	100n	22n	2n2	1n
f(Hz)	89	562	3548	22387
Q	4'16	4'16	4'16	4'16
A	1'4	1'4	1'4	1'4
R40	53K6	38K3	60K4	21K
R41	2K21	1K62	2K55	887 $\Omega$
R42	147K	107K	169K	59K
C27,C28	100n	22n	2n2	1n
f(Hz)	80	500	3K15	20K
Q	4'16	4'16	4'16	4'16
A	1	1	1	1
R43	82K5	59K	95K3	32K7
R44	2K49	1K82	2K87	1K
R45	165K	118K	187K	66K5
C29,C30	100n	22n	2n2	1n

f(Hz)	112	708	4467	-
Q	4'16	4'16	4'16	-
A	1'4	1'4	1'4	-
R46	42K2	66K5	48K7	-
R47	1K78	2K8	2K05	-
R48	118K	137K	133K	-
C31, C32	100n	10n.	2n2	-
f(Hz)	89	562	3548	-
Q	4'16	4'16	4'16	-
A	1'4	1'4	1'4	-
R49	53K6	84K5	60K4	-
R50	2K26	3K57	2K55	-
R51	150K	237K	169K	-
C33, C34	100n	10n	2n2	-
f(Hz)	100	630	4K	-
Q	4'16	4'16	4'16	-
A	1	1	1	-
R52	66K5	105K	75K	-
R53	2K	3K16	2K26	-
R54	130K	210K	150K	-
C35, C36	100n	10n	2n2	-
f(Hz)	141	891	5623	-
Q	4'16	4'16	4'16	-
A	1'4	1'4	1'4	-
R55	33K2	53K6	38K3	-
R56	1K40	2K21	1K62	-
R57	93K1	147K	107K	-
C37, C38	100n	10n	2n2	-



f(Hz)	125	800	5K	-
Q	4'16	4'16	4'16	-
A	1	1	1	-
R58	52K3	82K5	59K	-
R59	1K58	2K49	1K82	-
R60	105K	165K	118K	-
C39,C40	100n	10n	2n2	-

### 8.- LOS RECTIFICADORES.

Hay 30 rectificadores idénticos, uno por cada filtro instalado. Cada rectificador está constituido por un amplificador operacional con un diodo en el bucle de realimentación. Esta combinación actúa como un diodo ideal sin ninguna tensión umbral. Haremos esta demostración de forma detallada:

Si se aplica a un circuito rectificador como el de la figura (15), una tensión senoidal cuyo valor de pico sea menor que la tensión umbral  $V_{\gamma}$  (0'6v.) la salida siempre será nula.

Para rectificar señales del orden de milivoltios es evidente que será necesario disminuir el valor de  $V_{\gamma}$ . Colocando un diodo en el lazo de realimentación de un amplificador operacional, la tensión umbral queda dividida por la ganancia en lazo abierto  $A_v$  del amplificador y con ello  $V_{\gamma}$  queda virtualmente eliminada y el diodo se aproxima a un componente rectificador ideal.

Si en la figura (16), la entrada  $V_e$  es positiva en por lo menos  $V_{\gamma}/A_v$ , entonces  $V'$  sobrepasa a  $V_{\gamma}$  y D conduce. Debido a la conexión vir

tual entre las entradas inversora y no inversora - (provocada por la realimentación con el diodo en estado de conducción)  $V_o = V_e$ . Por tanto, el circuito actúa como un seguidor de tensión para señales positivas (con exceso de aproximadamente  $0,6/10^5 = 6\mu V.$ , siendo  $A_v$  del orden de  $10^5$ ). Cuando  $V_i$  es negativa, D deja de conducir y no se suministra ninguna corriente a la carga, salvo la pequeña corriente de polarización inversa de saturación del diodo. De la figura (16) deducimos:

$$V' = A_v \cdot V_i, \text{ donde } A_v = 10^5$$

$$V_i = V'/A_v \tag{1}$$

$$V_e = V_i + V_o \tag{2}$$

Sustituyendo la ecuación (1) en la --

(2) :

$$V_e = \frac{V'}{A_v} + V_o$$

$$V_e = \frac{V_i + V_o}{A_v} + V_o$$

$$V_e = \frac{V_i}{A_v} + \frac{V_o}{A_v} + V_o$$

$$V_e = \frac{V_i}{A_v} + V_o \left( 1 + \frac{1}{A_v} \right)$$

Al estar el término  $A_v$  en el denominador y ser su valor muy grande respecto al numerádor, se puede considerar el valor de la fracción - muy pequeño y por tanto, despreciable. Con lo cual podemos decir que:

$$V_e = V_o$$

Una vez demostrada la función de este circuito diremos que su misión principal es la de rectificar en media onda la señal de salida de los filtros. El bucle de realimentación negativa contiene el cursor de P1 para permitir el ajuste de -

la ganancia. La relación entre P1 y R2 se elige de modo que la gama controlable sea de unos 10dB. Este margen es necesario para compensar las diferencias de tensión entre los filtros debidas a las tolerancias de los componentes.

Teniendo en cuenta que Vi es pequeña la Ve está prácticamente en el cursor de P1. La ganancia mínima se obtendrá cuando el cursor esté totalmente desplazado al punto A, que en este caso será la unidad, ya que prácticamente la Ve es la Vo (despreciando Vi). La ganancia máxima corresponde al cursor totalmente desplazado al punto B y tendrá que ser de 10dB.

Calcularemos a continuación los componentes comunes a todos los rectificadores:

$$A_v = \frac{R_2 + P_1}{R_2}$$

$$10 = 20 \log \frac{R_2 + P_1}{R_2}$$

$$10^{1/2} = \frac{R_2 + P_1}{R_2}$$

Si fijamos el valor de P1=500K y lo sustituimos en la fórmula anterior nos queda:

$$R_2 = \frac{500K}{10^{1/2} - 1} = 231K$$

La red R-C situada después de cada --rectificador funciona como un circuito de muestreo --bloqueo, con el fin de obtener una cierta inercia de la visualización con respecto a la modulación --real. El condensador de tipo electrolítico, se carga a través de R1 y se descarga luego por R1/P1/R2

Hay que tener en cuenta, que la duración de carga esta relacionada con la frecuencia central del filtro correspondiente, lo que significa que la resistencia de carga tiene un valor diferente en cada rectificador.

La resistencia de descarga (P1 + R2) es la misma para todos los rectificadores. Puesto que la resistencia de carga está en serie con P1 y R2 durante la descarga, el tiempo de ésta es algo mayor para los filtros bajos que para los más altos.

#### 8.1.- CALCULO DE LAS RESISTENCIAS DE CARGA.

Como dijimos antes la resistencia de carga tendrá un valor diferente para cada rectificador, en función de la frecuencia central del filtro correspondiente, lo que significa que tendremos que calcular 30 resistencias distintas.

Sabemos que el tiempo de carga de un circuito R-C viene determinado por la constante de tiempo ( $\tau$ ) del mismo, que es:

$$\tau = R.C$$

Conocida la frecuencia central de cada filtro, hallaremos el valor del periodo correspondiente mediante la fórmula:

$$T = \frac{1}{f}$$

Para poder considerar que el tiempo de carga ha llegado a su máximo valor, sin tener en cuenta el tiempo que tarda el circuito R-C en cargarse exponencialmente, diremos que:

$$\tau = 10 T = R \cdot C$$

de donde:

$$R = 10 T/C$$

Aplicando esta fórmula a cada una de las resistencias, obtenemos la siguiente tabla de valores (teniendo en cuenta, que el condensador es el mismo para todas y de valor  $C = 4\mu 7 \text{ F.}$ ):

Frecuencia central $f_0(\text{Hz})$	Periodo $T(\text{sg})$	Resistencia $R$
25	0'04	100K
32	0'03	68K
40	0'025	56K
50	0'02	47K
63	0'016	39K
80	0'1255	27K
100	0'01	22K
125	$8 \cdot 10^{-3}$	18K
160	$6'25 \cdot 10^{-3}$	15K
200	$5 \cdot 10^{-3}$	12K
250	$4 \cdot 10^{-3}$	10K
315	$3'17 \cdot 10^{-3}$	6K8
400	$2'5 \cdot 10^{-3}$	5K6
500	$2 \cdot 10^{-3}$	4K7
630	$1'58 \cdot 10^{-3}$	3K9
800	$1'25 \cdot 10^{-3}$	2K7
1K	$10^{-3}$	2K2
1K25	$8 \cdot 10^{-4}$	1K8
1K6	$6'25 \cdot 10^{-4}$	1K5
2K	$5 \cdot 10^{-4}$	1K2
2K5	$4 \cdot 10^{-4}$	1K

3K15	$3.17 \cdot 10^{-4}$	680 $\Omega$
4K	$2.5 \cdot 10^{-4}$	560 $\Omega$
5K	$2 \cdot 10^{-4}$	470 $\Omega$
6K3	$1.58 \cdot 10^{-4}$	390 $\Omega$
8K	$1.25 \cdot 10^{-4}$	270 $\Omega$
10K	$10^{-4}$	220 $\Omega$
12K5	$8 \cdot 10^{-5}$	180 $\Omega$
16K	$6.25 \cdot 10^{-5}$	150 $\Omega$
20K	$5 \cdot 10^{-5}$	120 $\Omega$

#### 10.- EL VISUALIZADOR DE DIODOS LED's.

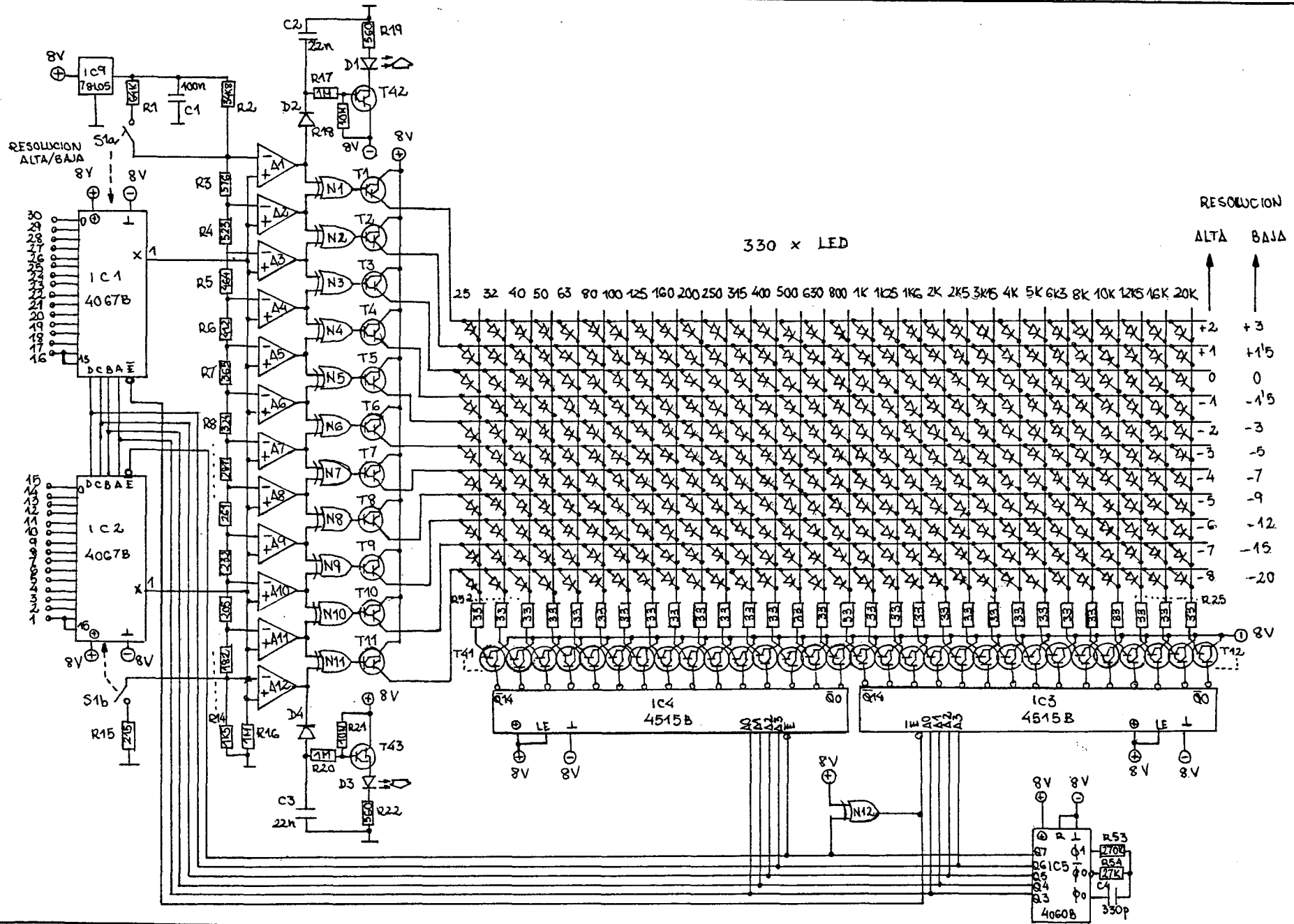
Está compuesto de 30 columnas de diodos LED para visualizar las tensiones de salida de los 30 filtros. Aparte de la matriz de 11 . 30 --- LED's está provisto de un dispositivo multiplexor/demultiplexor y los comparadores.

Los multiplexores de 16 a 1 (IC1,IC2) y los demultiplexores para las columnas 1 a 16 -- (IC3,IC4), están sincronizados por la señal de reloj procedente del oscilador/contador IC5, cuyas salidas Q3.....Q7 controlan las entradas A, B, C, D y  $\bar{E}$  (autorización) de los circuitos 4067 y las A0, A1, A2 y A3 de los 4515.

Se puede observar como la entrada x15 de los circuitos multiplexores no se utiliza, por lo que quedan 30 canales utilizables (tampoco se empleará, en consecuencia, la salida Q15 de los de multiplexores), como puede verse en la fig.(17).

El control alternado de los circuitos IC1/IC3 e IC2/IC4 se realiza con la ayuda de la señal Q7 de IC5 y de la puerta OR exclusiva N12 que

FIG. 17



que la invierte de modo que sólo uno de los dos multiplexores es activado cada vez. La red R53/R54/C4 determina la frecuencia del oscilador integrado de IC5, de modo que cada canal de los multiplexores permanezca activado durante 0'2ms.

Mientras que el multiplexor hace pasar la señal de salida de un filtro hacia la entrada de los comparadores el demultiplexor activa la columna correspondiente de la matriz, y así sucesivamente hasta recorrer las 30 columnas.

Para que una columna de la matriz sea activada, es preciso que una de las salidas de IC3 ó de IC4 pase al nivel lógico bajo, saturando el Darlington (T12....T41) correspondiente. Entonces, los cátodos de todos los diodos LED de esta columna estarán al potencial negativo de la alimentación. La corriente que circula a través de los LED's está limitada por R23....R52 y su valor de pico (debido al gran número de LED's que son multiplexados) se eleva a 300mA. por cada LED, aunque el medio es de sólo 10 mA. Eligiendo bien el LED y manteniendo alta la frecuencia de multiplexado evitamos que esto acorte demasiado la duración del LED.

Existen en el mercado relativamente pocos LED's que puedan soportar las corrientes de pico mencionadas pero se pueden conseguir con una corriente de pico de 1A. aproximadamente consultando los catálogos. En nuestro caso elegimos el diodo CQY 85Nb que cumple con este requisito.

El multiplexor IC1/IC2 tiene su salida aplicada a un comparador de 11 etapas, basado -



en A1...A12, cuyas entradas inversoras están polarizadas por un divisor de tensión de precisión (R2 ...R14). La tensión de referencia de este divisor (obtenida de IC9) es de 5 voltios, los escalones de comparación son de 1 dB y el nivel de 0 dB interno corresponde a una tensión de 0'5 voltios.

La entrada no inversora está alimentada por la señal de salida del filtro multiplexado en ese instante. A1...A12 comparan, pues, sucesivamente, la señal de salida de cada uno de los filtros con su correspondiente tensión de referencia. Cuando el potencial en la entrada no inversora es más elevado que el existente en la inversora, la salida del comparador pasa al nivel lógico alto, así como la entrada asociada de una de las puertas OR exclusivas (N1....N11). Las salidas de éstas controlan, a su vez, los transistores Darlington T1.....T11. Mediante esta combinación de las puertas OR exclusivas aseguramos que sólo se ilumine al tiempo un LED en cada columna, con lo que el consumo de corriente queda dentro de unos límites razonables.

Si las señales aplicadas al visualizador exceden su capacidad, se ilumina uno de los dos LED's auxiliares. El Darlington T42 hará que se active el LED D1 si la señal excede por encima. El LED D3 se ilumina, a través de T43, si queda por debajo o si no existe ninguna señal.

En cada caja se incluye un condensador y un diodo para mantener iluminado el LED un tiempo suficiente como para que se le vea.

La resolución del visualizador puede conmutarse a un margen menos preciso por medio de S1. Al cerrar este interruptor ponemos una resistencia adicional en paralelo con las resistencias superior e inferior de la cadena del divisor. El campo de medida comprende entonces desde +3 a -20 dB en lugar de +2 a -8 dB, cuando S1 está abierto.

#### 10.- GENERADOR DE RUIDO ROSA.

Aunque con la configuración vista hasta ahora nuestro analizador sería operativo, faltaba todavía una opción que permitiera efectuar medidas en todo el espectro audible: el generador de ruido rosa. Cuaquier señal que llegue a la entrada es objeto de análisis y luego se lleva al visualizador. No obstante, para la medición real se necesita una señal que contenga todas las frecuencias de la banda de audio que ha de medirse. De esta forma se logra una salida de lectura directa de las treinta bandas de frecuencia al mismo tiempo. La señal más adecuada para esta medición es el ruido rosa.

##### 10.1.- RUIDO ROSA.

Todo ruido posee un espectro aleatorio de frecuencias. En el caso en que todas las frecuencias estén igualmente representadas en la señal de ruido, éste recibe la denominación de ruido blanco. Podría parecer una señal de medida ideal para un analizador pero no es posible utili-

zarla con nuestro diseño. En efecto, resulta que - en los filtros de octava y de  $1/3$  de octava (estos últimos se emplean en nuestro analizador) el factor  $Q$  de todos es el mismo. Esto significa que el ancho de banda de cada filtro depende de su frecuencia central. Si utilizáramos ruido blanco la tensión de salida de cada filtro aumentaría con la frecuencia central porque el ancho de banda es mayor en las bandas de más alta frecuencia, la frecuencia medida se incrementaría en 3 dB por octava. Por consiguiente, el generador de ruido blanco debe ir seguido por un filtro pasabaja con una pendiente de 3 dB/octava para obtener una salida de lectura recta. Este ruido blanco filtrado se conoce como ruido rosa.

#### 11.2.- EL GENERADOR DE RUIDO.

Hay dos formas principales de generar ruido por medios electrónicos. En una de ellas se utiliza una unión de transistor ruidosa y en la otra una fuente de ruido digital. Nosotros emplearemos el segundo método, ya que proporciona resultados más fiables.

Nuestro generador de ruido está basado en un registro de desplazamiento con un periodo tal que la salida pseudo-aleatoria generada tiene un tiempo de repetición bastante largo y también parece "aleatorio". La disposición general del circuito se muestra en la figura (18).

El oscilador basado en  $N1/N2$ , que tiene una frecuencia de 1'5 MHz, suministra la señal

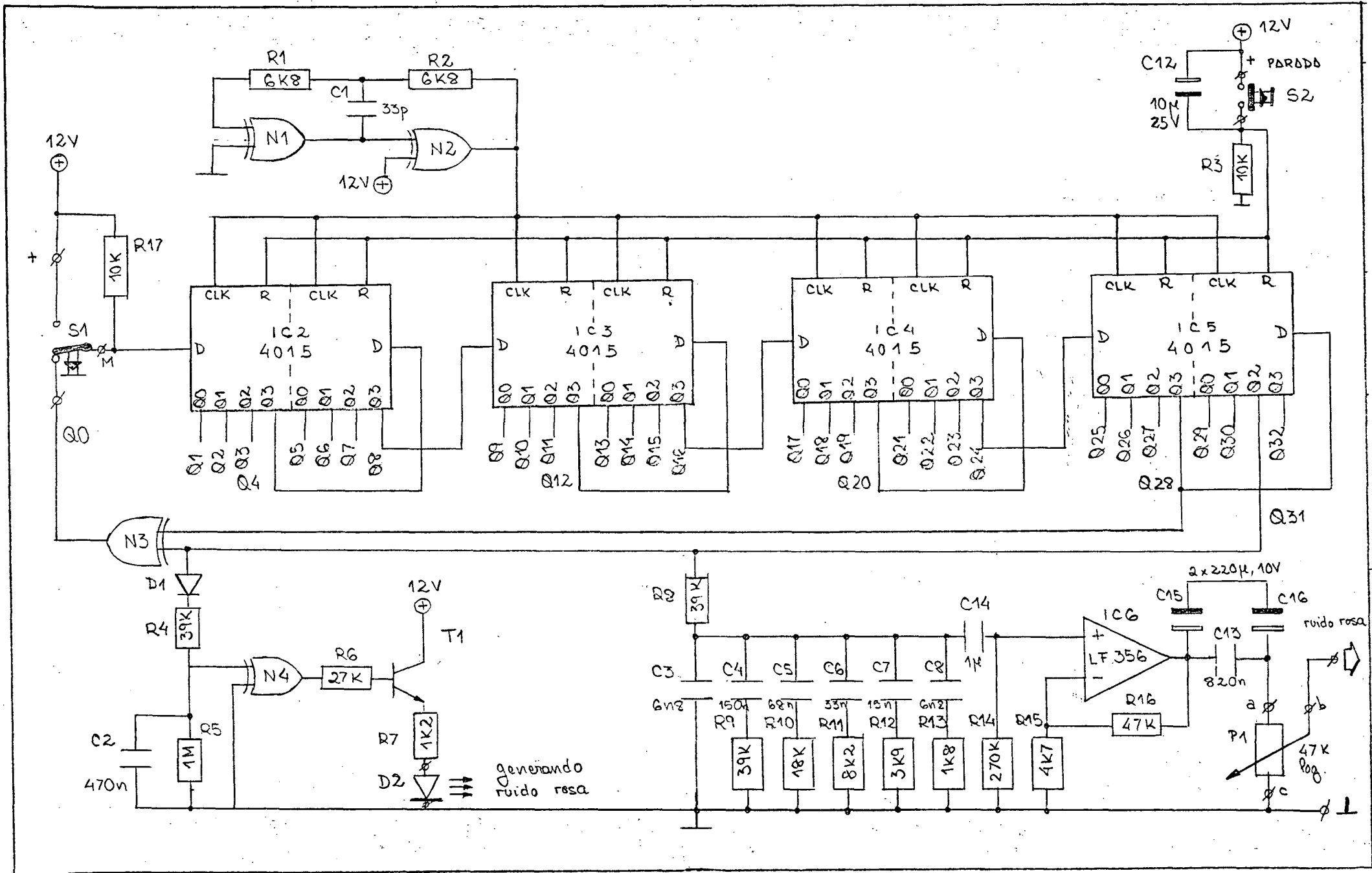
de reloj para el registro de desplazamiento de 31 bits, constituido por IC2...IC5. Las salidas Q28 y Q31 se realimentan a la entrada del registro de -- desplazamiento a través de la puerta OR exclusiva N3, con el resultado de que un ciclo comprende  $2^{31} - 1$  (=2147483647) impulsos de reloj. Con la frecuencia de reloj de 1'5 MHz utilizada en este caso, el ciclo del registro de desplazamiento es de unos 25 minutos, por lo que podemos hablar realmente de -- que el ruido es aleatorio.

El inconveniente de este registro de desplazamiento es que nunca puede alcanzarse la situación en la cual el registro contenga todos "ceros" porque, entonces, el circuito se mantendría - inactivo. Este problema se resuelve con facilidad añadiendo dos pulsadores que ponen en marcha y detienen respectivamente el registro de desplazamiento. El pulsador de marcha (S1) produce la aparición de un número cualquiera de bits al nivel lógico alto, con una configuración de origen del ciclo de unos 2000 impulsos. El generador se para al pulsar S2. A continuación, el registro de desplazamiento se llena por completo de "ceros" y se mantiene en esa situación.

Hemos incluido un circuito de indicación para mostrar cuando el generador está proporcionando ruido rosa. Siempre que aparezca un "1" - en la salida Q31 el condensador C2 se carga a través de D1 y R4. Antes de que se descargue este condensador T1 dispone de tiempo para entrar en saturación gracias a N4 y encender el LED D2, con lo - que nos indica que estamos en presencia de una se-

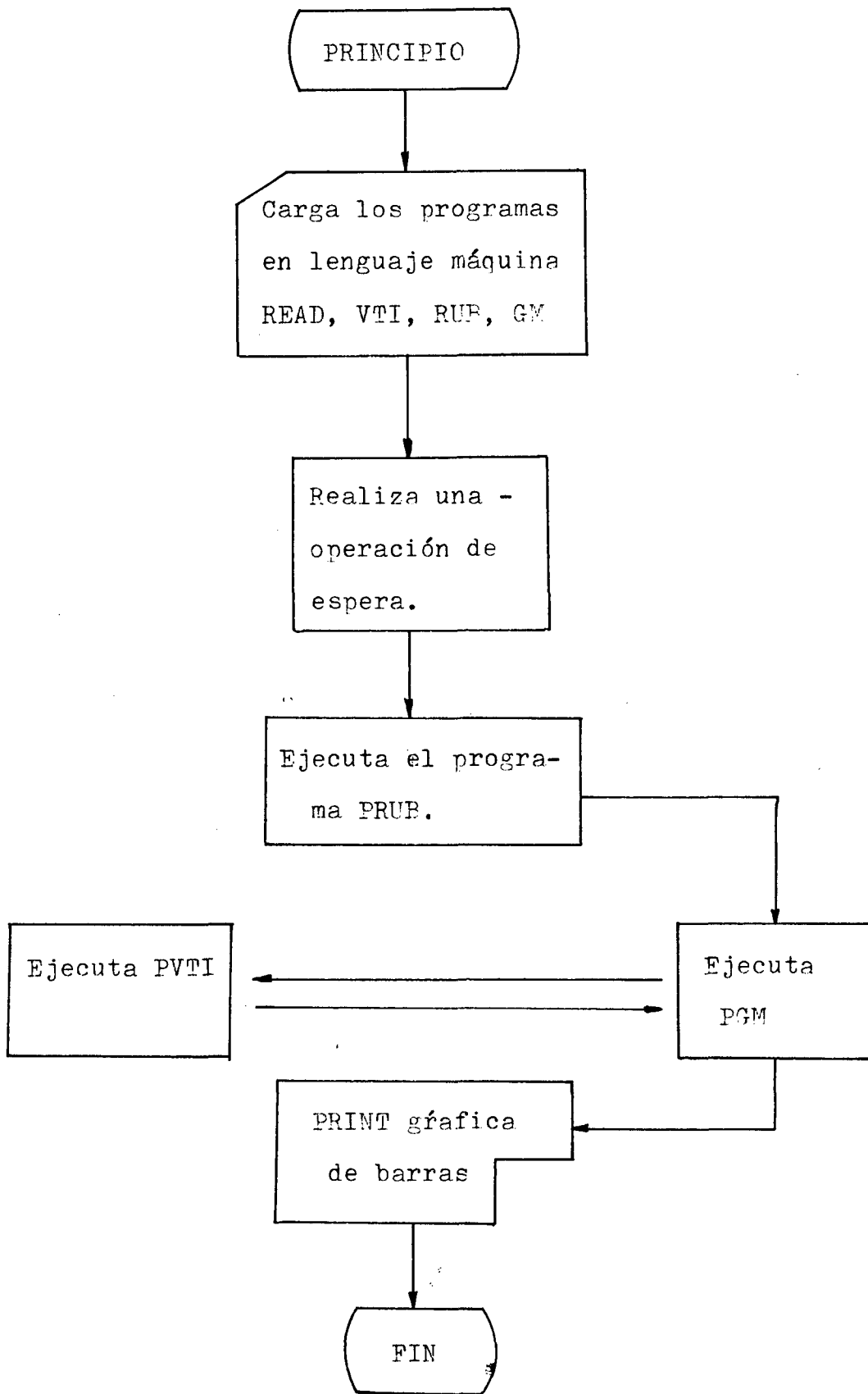
ñal de ruido rosa. Los tiempos de carga y descarga de C2 se eligen de modo que el LED se ilumine de forma permanente cuando el generador esté funcionando. La señal de ruido blanco que aparece en la salida Q31 (patilla 11 de IC5) se filtra por medio de una red de ruido rosa constituida por R8...R13 y C3...C8, cuyo conjunto forma un filtro Tchebyshev de seis etapas con una desviación teórica inferior a 0'14 dB con respecto a la línea de -3 dB/octava entre 12'3 Hz y 31'5 KHz. En la práctica, esto significa que la desviación con respecto a la línea ideal depende solamente de las tolerancias de los componentes utilizados en la red.

El filtro va seguido por un amplificador separador (IC6) cuya ganancia se ajusta a 11. El potenciómetro P1 se emplea para controlar la señal de salida. Los condensadores C13, C15 y C16 hace que esta señal no contenga ninguna componente de corriente continua.



# 11.- VISUALIZACION POR ORDENADOR

## DIAGRAMA DE FLUJO



## 11.2.- ALGORITMO.-

- 1<sup>o</sup> Lee los programas en lenguaje máquina y los va colocando en unos lugares determinados.  
READ ( PVIT, PRUB, PGM)
- 2<sup>o</sup> El programa espera un tiempo que se le da por el teclado, debido al tiempo que tiene que estar conectado el READ(T).
- 3<sup>o</sup> Ejecuta el programa reubicador colocando el -- PVIT en la posición &H66, que es el salto de la NMI (interrupción no enmascarable).
- 4<sup>o</sup> Ejecuta el programa de gobierno del multiplexor y durante la ejecución de este programa se produce la interrupción cargando las amplitudes en memoria.
- 5<sup>o</sup> Dibuja la gráfica de barras dando la altura, - el contenido en memoria.  
WRITED (GRAFICA)

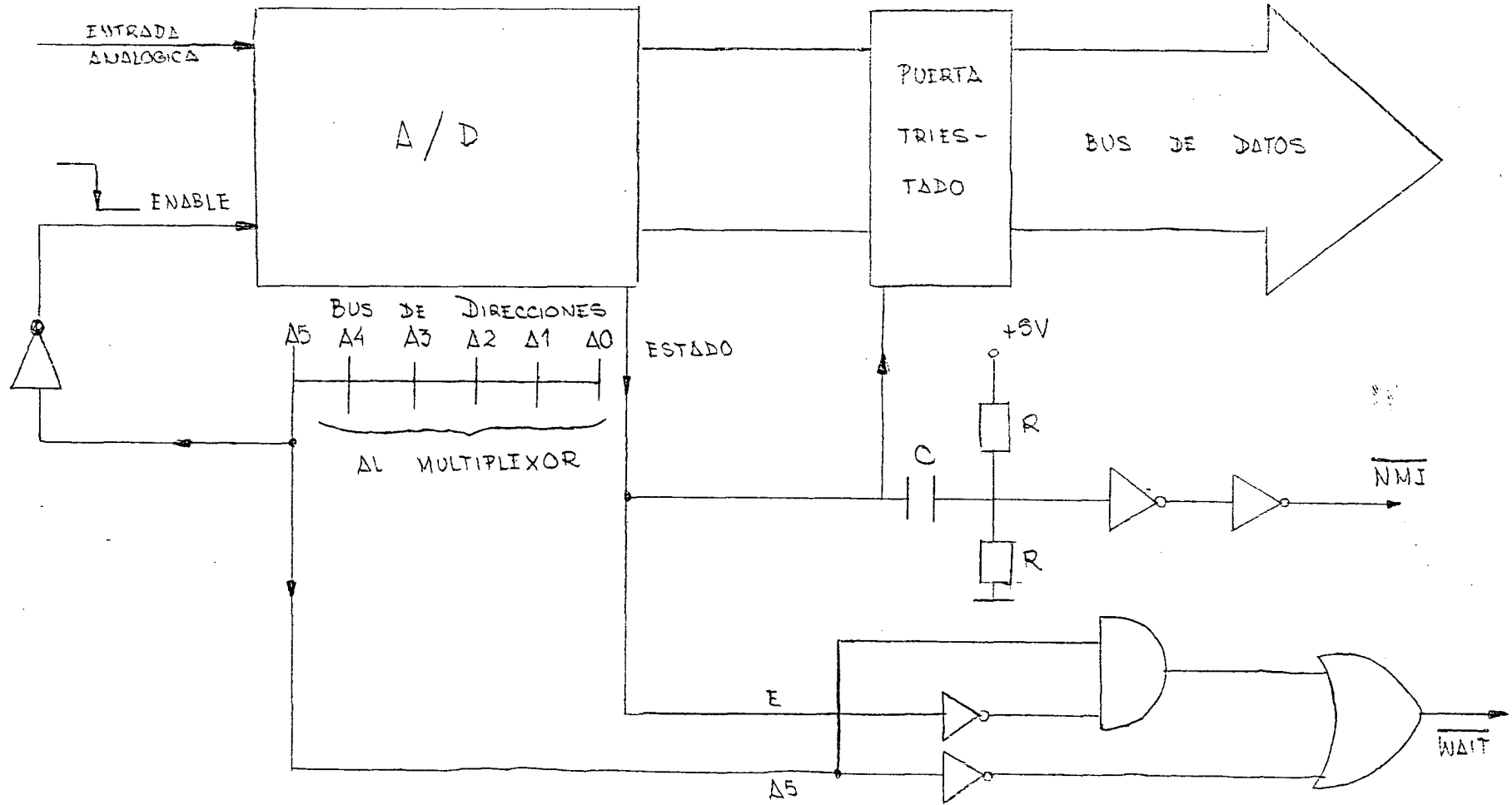
## 11.3.- ESQUEMA DE BLOQUE DEL CONVERTIDOR A/D.

Pasaremos a explicar a continuación - el esquema de bloques que presentamos en la siguiente página.

El convertidor A/D nosea una salida - de estado, por la cual nos presenta el dato disponible con el valor 0 ó en caso contrario con el valor 1 (dato no disponible). En este último caso desactiva el triestado para que el resultado pase al bus de datos.

El nivel bajo en la salida de estado provoca la aparición de un impulso negativo a la entrada del conjunto de inversores, lo que provoca





a la salida del mismo el impulso negativo que activa la  $\overline{NMI}$  (activa a nivel bajo). Esto sucede del siguiente modo:

La entrada del primer inversor, por medio del divisor de tensión, tiene 2'5 voltios -- que es interpretado a la salida del segundo como un 1, el impulso negativo reduce estos 2'5 voltios en el primer inversor lo que permite tener a la salida del segundo el 1 anteriormente mencionado.

El diferenciador se coloca a la salida de estado para provocar un impulso de corta duración que dure menos que el programa de interrupción.

El valor de la constante de tiempo -- que determinará la duración del impulso vendrá dado por la fórmula siguiente:

$$\tau = R/2 \cdot C$$

Veamos cual es su valor matemático:

Los ciclos de reloj correspondientes a las sentencias empleadas en la interrupción  $\overline{NMI}$  son los siguientes.

Sentencias	Nº de ciclos de reloj
CALL	17
PUSH qq	11
LDA (mn)	13
INIC	6
PUSH	11
PUSH	11
LDA, 3	9
CALL	17
POP HL	10

POP HF	10
LD(HL),A	7
POP AF	10
CALL	17
RET	<u>10</u>

159 ciclos de re

loj.

El Z80 trabaja con una frecuencia de reloj de 3'58MHz, es decir que en un segundo hay - 3'58.10<sup>6</sup> ciclos. Tenemos que calcular el número de segundos que hay en 159 ciclos:

$$\text{seg.} = \frac{159}{3'58 \cdot 10^6} = 44'41 \mu\text{seg}$$

Este valor corresponde a la duración de PVTI. Haciendo que la constante de tiempo sea de 30μseg. conseguimos que el impulso sea de corta duración con respecto al de la ejecución del programa PVTI. Fijando el condensador C=10μF el valor de las resistencias serán de:

$$R = \frac{2\tau}{C} = 6\Omega$$

Siguiendo el diagrama de bloques podemos ver también el bus de direcciones que va al -- multiplexor, siguiendo la siguiente regla:

A5	A4	A3	A2	A1	A0	
1	0	0	0	0	0	= 0
1	0	0	0	0	1	= 1
⋮						
⋮						
1	1	1	1	1	1	= 29

La transición del bit A5 de 0 a 1 pro

voca la activación del convertidor A/D. Cuando no se direcciona el bus su valor es 0, en el caso contrario su valor es 1.

La señal  $\overline{\text{WAIT}}$  para la CPU cuando su valor es 0. Esta señal tiene que cumplir la siguiente tabla:

ESTADO	A5	$\overline{\text{WAIT}}$	
1	1	0	: se para - la CPU.
1	0	1	: no se pa- ra la CPU
0	1	1	
0	0	1	

La CPU se para congelar la dirección del multiplexor en el filtro que está analizando, cuando el dato es analizado pasa a la siguiente posición (dirección) del multiplexor.

Para que pueda trabajar bien el convertidor se coloca un divisor de frecuencia a la salida del reloj del Z80, ya que es muy alta para el reloj del convertidor A/D.

PROGRAMA

```
10  REM CARGA DEL PROGRAMA VECTOR TRATAMIENTO DE
    LA INTERRUPCION
20  CLEAR, &H8FFF
30  FOR X=&H9000 TO &H9015
40  READ N: POKE X,N: NEXT X
50  DATA ( PROGRAMA ; EN LENGUAJE MAQUINA, PVIT)
60  REM CARGA PROGRAMA REUBICADOR
70  FOR X=&H901F TO &H903F
80  READ N: POKE X,N: NEXT X
90  DATA (PROGRAMA, EN LENGUAJE MAQUINA, PRUP)
100 REM CARGA PROGRAMA DE GOBIERNO DEL MULTIPLE--
    XOR
110 FOR X=&H9049 TO &H9075
120 READ N, POKE X,N: NEXT X
130 DATA (PROGRAMA, EN LENGUAJE MAQUINA, PGM)
140 REM PROGRAMA DE MUESTREO
150 REM T=n.50; n=N0 DE SEGUNDOS
160 TIME=0: INPUT "¿TIEMPO?"; T
170 IF TIME=T THEN 200
180 GOTO 170
200 DEF USRO=&H901F: DEF USR1=&H9049
210 X=USRO (I): REM EJECUTA EL PROGRAMA DE REUPI-
    CACION
220 X=USR1 (I): EJECUTA EL PROGRAMA DE MUESTREO
230 REM PROGRAMA REPRESENTACION GRAFICA DE MUES-
    TREC
240 SCREEN 2
250 FOR Y=0 TO 180 STEP 9
260 LINE(0,Y)-(255,Y): NEXT Y • DIBUJA CARATULA
270 OPEN "GRP": "AS#1
```

```
280  PRESET (30,183)
290  PRINT #,1"ESPECTRO DE FRECUENCIAS"
295  CLOSE #1
300  MAXFILES 15
310  FOR Y=1 TO 15: OPEN "GRP ":"AS#Y: PRESET (0,
      173-(Y-1).9): PRINT#Y,Y-1:CLOSE#Y: NEXT Y:
      REM ROTULA LA ESCALA
320  FOR X=30 TO 233 STEP 7
330  LINE (X,180)-(X+5,180-(PEEK(&H9072+X-30)/255
      . 180),3,BF: NEXT X: REM DIBUJA COLUMNAS
340  GOTO 340
350  END
315  MAXFILE 5
316  FOR Y=1 TO 5: OPEN"GRP"AS#Y: PRESET(0,173-(
      14+Y).9):PRINT#Y, 14+Y: CLOSE#Y: NEXT Y: REM
      ROTULA ESCALA
```

Comentarios:

- 20 Crea espacios para programar en código máquina entre las direcciones H8FFF y HF380. La memoria comprendida entre las direcciones H8FFF y HF380 están consideradas como área de trabajo del sistema por la ROM dedicada al BASIC.
- 30 Desde la posición H9000 TO H9015, la instrucción POKE escribe un byte en cada posición de memoria. Carga el programa en código máquina.
- 200 Define la dirección de comienzo de la subrutina en código máquina. La 0 comienza en 901FH y la 1 en 9049H.

210    USR llama al programa en código máquina, que comienza en la posición especificada por la instrucción DEF USR. Para llamar a una subrutina en código máquina el ordenador debe de saber la dirección de comienzo de la subrutina.

240    SCREEN 2 se emplea para fijar el modo de dibujar en la pantalla. El 2 significa que es modo de alta resolución. Tiene 256 puntos de anchura por 192 de largo. Para determinar un punto en la pantalla se hace del siguiente modo:

                  El punto de la parte superior izquierda es la posición de coordenadas (0,0); el punto de la parte inferior derecha se corresponde a (255,191).

0	SCREEN 2	255
0		X
191		
Y		

                  Los grupos en esta pantalla se ordenan en grupos de 8 puntos de longitud. El primer grupo de puntos se encuentra entre las coordenadas (0,0) y (8,0). Cada grupo de 8 puntos sólo puede contener dos colores, el color de texto y el color de fondo. Grupos adyacentes pueden contener colores distintos. La resolución del color para la pantalla es de 33.192.

260    LINE esta instrucción de gráficos dibuja rec

- tas y rectángulos. Estos podrán ir coloreados. LINE (0,Y)-(255,Y) mediante un bucle de  $\Delta$  STEP 9 dibuja una caratula de 20 lineas.
- 270 Para escribir caracteres en modo gráfico se ha de abrir un fichero para las pantallas --gráficas "GRP:" , con un número de fichero, #1.
- 280 PRESET (30,183) el cursor gráfico se posiciona en el punto (30,183).
- 290 Escribe en la pantalla gráfica #Espectro de Frecuencias".
- 300 Pone escala a la caratula.
- 310 PEEK devuelve el byte contenido en la posición de memoria especificada. Dibuja las treinta columnas en barras de amplitudes correspondientes a la información codificada por el convertidor A/D.
- 315 Ya que el número máximo de ficheros que se puede leer a la vez es 15, tenemos que volver a fijar 5 más para poder terminar de rotular la escala.
- 330 Dibuja barras coloreadas en verde.  
(X,180) son las coordenadas de la esquina inferior del rectángulo.  
X+5 es la coordenada X de la otra esquina --del rectángulo.  
PEEK lee el contenido de memoria en la dirección donde están guardadas las amplitudes, --que para X=30 está en la dirección más baja &H9072. La máxima amplitud corresponde a 255



(B11111111); al dividir por 255 y multipli--  
car por 180 hacemos que la máxima amplitud -  
corresponda a la dimensión de la pantalla.

El número 3 corresponde al color verde y BF  
que es una barra(para diferenciarlo de una -  
recta).

340 Mantiene en la pantalla la gráfica.

160 TIME esta función devuelve el valor del re-  
loj interno al sistema. Al conectar el orde-  
nador se inicializa a 0 y se incrementa cada  
vez que el procesador de la pantalla de vi--  
deo (VDP) hace una interrupción; esto sucede  
50 veces por segundo. Mediante TIME le pode-  
mos dar cualquier valor al reloj del sistema  
TIME=0 lo que hace es poner a 0 el reloj del  
sistema. Como se incrementa 50 veces en un -  
segundo, si queremos que el programa tarde 1  
hora pondremos  $T=60.50=3000$ . Este valor se -  
lo damos al ordenador.

170 Se compara T con el reloj, si no coincide es  
perará el programa hasta que suceda.

PROGRAMA DEL GOBIERNO DEL MULTIPLEXOR Y EL  
CONVERTIDOR A/D. (PGM)

DIRECCION CODIGO

&H4049	CD	CALL &H0138: lee registro selecc ción de slot.
&H905A	38	
&H905E	01	
&H905C	F5	PUSH AF: Apila el contenido del acumulador.
&H905D	3E	LDA,2: Carga el número de slot- 2, en este caso bus de - expansión.
&H905E	02	
&H905F	26	LDH,&H0: Carga H con 0 para se- leccionar la página 0 - que junto con A-2 selecc ciona página ) del slot 2 (Bus de expansión).
&H9060	00	
&H9061	CD	
&H9062	24	CALL, &H0024:Selecciona página 0 del slot 2.
&H9063	00	
&H9064	21	LD HL, &H9071:Carga la direcc ión menor-1 donde se alojarán las amplitudes en - la memoria dados por el convert tador.
&H9065	71	
&H9066	90	

&H9067	FB	
&H9068	11	
&H9069	20	EI:activa las interrupcio-- nes, ya que la subrutina situada en '&H0024 las de sactiva.
&H906A	00	
&H906B	47	LD DE,&B100000·Guarda en DF el valor de dirección -- del 1 <sup>o</sup> multiplexor.
&H906C	1E	
&H906D	13	LD B,30:número de amplitu-- des a guardar.
&H906E	12	INC DE (\$)·Incfementa en 1 el valor de DE para apuntar el siguien-- te multiplexor.
&H906F	10	
&H9070	FD	
&H9071	F1	LD (DE),A:Apunta por el bus de direcciones al multip. y al conv.
&H9072	CD	
&H9073	3B	
&H9074	01	DJNZ,FD:Salto a(\$ <u>)</u> siempre que R≠0. Si R=0 sal ta a la siguiente - instrucción.
&H9075	C9	PCP,AF·Recupera el registro de selección de slot primitivo.  CALL &H0EB·Devuelve el re--

gistro primitivo.

RET. Regresando al sistema.

Comentarios:

CALL &H0138 Es la llamada a una subrutina. Esta subrutina lee el contenido del registro de selección de slot, que se encuentra en la posición &H48. El acumulador contendrá una copia del registro de selección de slot. La ejecución de esta instrucción cambia el contenido del acumulador.

CALL &H0024 Esta llamada selecciona una página de un slot. Los dos bits más significativos del registro H se emplean para seleccionar la página correspondiente, del siguiente modo:

BUS	Página selecc.
00	0000-3FFF pág. 0
01	4000-7FFF pág. 1
10	8000-BFFF pág. 2
11	C000-EFFF pág. 3

El acumulador indica el slot a seleccionar. Esta llamada altera los contenidos de AF, PC, DE y HL. Esta subrutina desactiva todo tipo de interrupciones.

LD (DE),A Carga el contenido de A en la posición que contiene DE. Esto se hace así para que el bus direcciona el multiplexor.

DJNZ d      Decrementa el registro B. Si B=0 Pc+Pc+d  
Si B=0 continúa con la siguiente instrucción.

CALL &H013B      Con esta llamada escribimos en el registro de selección de slot. El acumulador ha de tener el valor a escribir. No afecta a ningún registro, ni tampoco a posición alguna.

RET              Retorno del subprograma.

PROGRAMA DE TRATAMIENTO DE INTERFUNCIONES (PVTI) Y CARGA EN MEMORIA LAS AMPLITUDES CODIFICADAS POR EL CONVERTIDOR A/D.

DIRECCION      CODIGO

&H66	CD	CALL&H0138:lee el registro de selección de slot.
&H67	38	
&H68	01	
&H69	F5	PUSH AF:Apila el registro leído.
&H6A	3A	LDA(&H0):Carga lo que hay en el bus de datos en A
&H6B	00	
&H6C	00	
&H6D	23	INC HL:Incrementa en 1 el contenido de HL.
&H6E	F5	PUSH AF:Apila la amplitud obtenida del conv. A/D.
&H6F	E5	PUSH HL:Apila la dirección de memoria donde guarda la

muestra.

&H70	3E	LDA,3:Carga el slot 3.
&H71	03	
&H72	CD	CALL &H0024·Selecciona slot.
&H73	24	
&H74	00	
&H75	E1	POP HL:Desapila la dirección don de se encuentra la ampl.
&H76	F1	POP AF:Desapila la amplitud.
&H77	77	LD (HL),A:Carga en memoria la am plitud A/D.
&H78	F1	POP AF·Desapila el registro de - selección de slot de en-- trada.
&H79	CD	CALL &H013B:Devuelve el registro primitivo (slot 2).
&H7A	3B	
&H7B	01	
&H7C	C9	RET·Regresa al sistema.

Comentario·

INC HL· Se incrementa porque en HL está la direcci  
ón donde se guardan las amplitudes en memo  
ria (HL---HL+1)

PROGRAMA REUBICADOR (PREU): TRASLADA EL VECTOR DE  
TRATAMIENTO DE LAS INTERRUPCIONES DE &H9000 A &H66  
(DIRECCION DONDE SE DIRIGE LA CPU AL RECIBIR NMI)

DIRECCION	CODIGO	
&H901F	CD	CALL &H0138:Lee registro se- lección de slot.

&H9020	38	
&H9021	01	
&H9022	F5	PUSH AF:Apila el registro anterior.
&H9023	21	LD HL,&H9000·Carga 1 <sup>o</sup> dirección programa a trasladar.
&H9024	00	
&H9025	90	
&H9026	11	LD DE, &H66·Carga 1 <sup>o</sup> dirección lugar a trasladar.
&H9027	66	
&H9028	00	
&H9029	01	LD BC, &H16·Carga número de instrucciones.
&H902A	16	
&H902B	00	
&H902C	B5	PUSH HL } &H16:Apila los tres registros. PUSH DE } PUSH BG }
&H902D	D5	
&H902E	C5	
&H902F	3E	LDA, 3·Carga el slot 3 (memoria)
&H9030	03	
&H9031	F3	DI·Desactiva interrupciones.
&H9032	CD	CALL &H0024:Selección de slot.
&H9033	24	
&H9034	00	
&H9035	C1	POP BC } :Desapila registros salvados POP DE } POP HL }
&H9036	D1	
&H9037	E1	
&H9038	ED	LDIR:Realiza el traslado de bytes desde &H9000 a &H66 e

en adelante.

&H9039	BO	
&H903A	F1	POP AF·Desapila registro selección de slot.
&H903B	CD	CALL &H013B·Devuelve el valor primitivo al slot.
&H903C	3B	
&H903D	01	
&H903E	FB	EI·Activa las interrupciones.
&H903F	FC9	RET retorno de subrutina.

Comentarios:

La instrucción LDIR transfiere el octeto situado en la dirección HL a la posición de memoria situada en la posición de memoria DE. DE y HL son incrementados mientras que BC es decrementado. Esto se repetirá hasta que BC sea igual a cero.

BC desarrolla así la función de contador de bucle, inicializando BC a la cantidad de octetos que se desea desplazar (en nuestro caso &H16), HL a la dirección inicial de la zona a desplazar (&H9000) y DE con la dirección inicial de la zona destino, utilizando la instrucción LDIR, el número de octetos deseados se desplaza de la zona inicial a la zona final. Esquemáticamente será:

(DE)=====	(HL)	} Repetir mientras BC≠0
(DE)=====	(DE+1)	
(HL)=====	(HL+1)	
(BC)=====	(BC-1)	



## APENDICE I.-

### TRATAMIENTO DE LOS COMPONENTES

Antes de iniciar el montaje de los circuitos impresos de cualquier equipo electrónico es preciso que los componentes que van a ser montados reciban una preparación previa que facilite al máximo su inserción en los taladros correspondientes, así como su posterior soldadura.

Las operaciones necesarias podrán realizarse a todo el conjunto completo de componentes antes del montaje de cualquiera de ellos o durante el mismo, preparando cada uno individualmente y pasando, acto seguido, a su inserción en el circuito.

Esta preparación previa se realiza con objeto de adaptar, de la mejor manera posible, la forma y dimensiones de cada componente al espacio físico de que va a disponer sobre el circuito, empleándose para todo ello la denominación de preformado. Aunque todas estas operaciones puedan parecer secundarias, dependiendo del punto de vista de cada persona, son muy recomendables si se desean evitar riesgos de roturas de terminales, cortocircuitos accidentales, daños por la temperatura que pueden alcanzar algunos elementos, obteniéndose un circuito terminado en el que no existirá ningún problema de identificación del valor de cada componente, con lo que se facilitan al máximo las posibles reparaciones posteriores. También se incluye dentro del concepto de preformado el corte de los terminales si se opta por realizarse antes de la soldadura. Los

hilos y cables deberán ser cortados a la medida necesaria y pelados en los extremos de conexión, operación también incluida en esta fase previa al montaje.

### I.1.- COMPONENTES DE DOS TERMINALES.

Los componentes que únicamente disponen de dos terminales pueden ser clasificados en dos grupos, atendiendo a la forma empleada para la colocación de estos sobre el cuerpo durante el proceso de fabricación. Son los siguientes:

- .- Componentes con terminales axiales.
- .- Componentes con terminales radiales.

Los primeros presentan los terminales de salida situados sobre los extremos del cuerpo y alineados con éste, formando una línea imaginaria que pasaría por su centro geométrico. Requiere, por lo tanto, que se les realice un preformado para su montaje en circuito impreso.

En el segundo grupo, los terminales de los extremos del cuerpo son perpendiculares a éste y paralelos entre sí. Se adaptarán, sin preformado, al montaje sobre el circuito impreso si la distancia entre los tralados en que deban ser incertados coincide con la separación entre los terminales. En caso contrario necesitarán un preformado para su adaptación al circuito.

Para realizar esta operación es necesario conocer previamente a qué distancia deben de doblarse los terminales para su incersión en el circuito, ello requiere realizar la medida de la dis--

tancia entre los taladros de montaje mediante un calibre o con una regla graduada en milímetros, ya que no se requiere una precisión alta. Los componentes con terminales axiales se montarán paralelos al circuito impreso y generalmente apoyados en él, en su preformado deberá procurarse que la referencia o valor que aparezca en el cuerpo quede visible para facilitar una rápida identificación después de su inserción.

## I.2.- CONFORMADO DE TERMINALES.

El doblado a medida de los terminales puede ser realizado a mano o con algunas herramientas especiales para este trabajo.

En el doblado a mano se tendrá presente un conjunto de precauciones que eliminarán el riesgo de rotura del terminal o del cuerpo del componente. A continuación se enumera las principales:

.- El doblado se realizará con un alicate con puntas finas haciendo presión en el punto de doblado lo que permite aislar del cuerpo del componente las presiones que se apliquen sobre el terminal.

.- No se ejercerá fuerza sobre la zona de unión del cuerpo con el terminal, ya que podría desprenderse éste.

.- El doblado no debe de quedar con un exagerado ángulo recto, sino que se procurará formar una pequeña curvatura para que el alambre de cobre del terminal no se quiebre en ese punto.

.- Se procurará realizar el preformado se realiza en forma simétrica con respecto al cuerpo.

.- Existe una herramienta especial para preformar, denominada conformador de componentes en la que una vez ajustada la distancia entre los dos puntos de doblado, realiza toda la operación -- con una única manipulación, de tal manera que por su forma de trabajo, evita los riesgos mencionados anteriormente. Esta herramienta es muy útil sobre todo cuando existe un cierto número de componentes que deban ser preformados a la misma distancia, ahorrando, de esta manera, una considerable cantidad de tiempo.

Los componentes con terminales de salida radiales se pueden montar directamente y presentan generalmente sus terminales ya cortados a la longitud adecuada, con lo que se evita la necesidad de realizar cortes posteriores a la soldadura, sin embargo, a veces es preciso realizar un conformado, cuando la distancia entre taladros del circuito -- impreso, para ello y una vez conocida esta separación, se efectúa manualmente sobre cada terminal un primer doblado, con lo que ambos adoptarán una forma similar a los axiales. En este momento ya puede ser fijada la distancia necesaria, realizándose un segundo y definitivo doblado, con el que ya se obtiene la forma final que el circuito requiere. Deben ser tenidas en cuenta las recomendaciones mencionadas anteriormente para evitar daños en el componente.

Existe también una herramienta, espe--

cial para esta disposición de terminales que realiza todas las operaciones citadas con una sola manipulación, presentando la misma ventaja de ahorro de tiempo. Los componentes con un mayor número de terminales también requieren un conformado previo al montaje, aunque en este caso, la necesidad aparece como consecuencia de que éstos suelen llegar al usuario doblados o deformados impidiendo el montaje inmediato y directo sobre el circuito. Entonces se -- precisa volver a llevarles a su posición original, utilizándose para ello el alicate de puntas finas, con el que se sujetará el terminal por un punto, lo más alejado posible de la unión con el cuerpo del -- componente, realizando con la mano u otro alicate -- todas las operaciones de enderezado del hilo, hasta que pueda ser insertado sin dificultad en los tala-- dros que le correspondan.

### I.3.- ESTAÑADO PREVIO.

En ocasiones y siempre que se tenga du-- da sobre la buena soldabilidad de los terminales -- por presentar éstos un aspecto sucio u oxidado, se-- rá necesario realizarles un estañado previo al mon-- taje, para lo que se deben sumergir en un crisol o -- cazoleta con aleación fundida de estaño-plomo, de -- los que existen en el comercio, adaptables o no al cuerpo del soldador. Si no se dispone de este acce-- sorio, podrá realizarse el estañado directamente -- con la punta del soldador aportando una pequeña can tidad de hilo de soldar que deberá distribuirse uni formemente sobre el terminal, procurando evitar la

aparición de zonas en que el depósito de estaño sea alto y presenten un espesor tal que impidan la inserción en el circuito impreso. El terminal se deberá sujetar durante toda la operación con un alicate situado entre la zona a estañar y el cuerpo del componente, con lo que se evitará un sobrecalentamiento de éste al comportarse el alicate como radiador térmico.

#### I.4.- COMPONENTES DE MAYOR POTENCIA.

Antes del montaje se separarán aquellos componentes que vayan a producir, durante el funcionamiento, una cierta radiación de calor por estar sometidos a unas condiciones de trabajo que les obligan a disipar una determinada potencia. El montaje de todos ellos se realizará al final, de forma que pueda tomarse un mínimo de precauciones para impedir que su temperatura pueda dañar a otros componentes cercanos. El caso más común será con las resistencias de una potencia superior a 0'5 vatios y con algunos transistores de media y alta potencia. Para todos ellos se respetará una separación mínima del circuito impreso de 5 milímetros, ya que en caso contrario resultaría seriamente dañado este. Respecto al resto de componentes se procurará fijar una separación mínima de 10 milímetros.

En el caso de algunas resistencias de alta potencia con un cuerpo relativamente grande y que presentan sus dos salidas por el mismo extremo del cuerpo, se utilizarán unos soportes metálicos especiales que aseguren su fijación mecánica e impidan cualquier movimiento que pueda llegar a rom--

per los terminales. Una vez efectuadas todas las operaciones descritas, se obtendrá un conjunto de componentes adaptados perfectamente al circuito impreso, pudiendose realizar su inserción en este sin ninguna dificultad. Con los hilos se deberá, también, tener en cuenta una serie de precauciones para su preparación e instalación en los circuitos y equipos.

#### I.5.- MONTAJE DE HILOS Y CABLES.

El hilo o alambre "desnudo" que se va a emplear para enlazar sobre el circuito impreso -- los puntos que lo requieran, se cortará con el alicate de corte, a la medida necesaria, efectuando un doblado en cada extremo de una manera análoga a la realizada con los terminales de componentes. Las -- porciones de alambre, así obtenidas y denominadas -- "puentes" quedarán aptas para su montaje, pudiendo procederse a realizar este sin mayor dificultad.

Los cablecillos necesarios para la interconexión entre circuitos impresos o entre estos con los componentes situados fuera de los mismos, -- deben ser cortados a la longitud precisa, eliminandose después un trozo de la cubierta aislante de -- los extremos, operación denominada "pelado", en una longitud de 4 o 5 milímetros aproximadamente. Para ello se utilizará un alicate o tenaza de pelar adecuado teniendo en cuenta las siguientes recomendaciones:

.- El alicate deberá estar perfectamente ajustado al diametro del conductor para que úni-

camente se corte la cubierta, evitando cortar este de forma inadvertida.

.- No deberá quedar dañado el conductor en ningún punto para evitar riesgos posteriores de roturas.

.- Se evitará dejar restos de la cubierta en la zona "pelada" con objeto de no tener ningún problema en la posterior soldadura.

En los cables apantallados y coaxiales una vez que se ha realizado el "pelado" de la cubierta externa, se separará la malla trenzada que forma el conductor exterior o "pantalla", agrupando todos los hilos de ésta en un punto y quedando al descubierto el conductor interno, protegido con su correspondiente cubierta. Sobre éste se realizará un segundo "pelado", procurando dejar una cierta longitud de cubierta que garantice el aislamiento del otro conductor. Si se trata de un cable paralelo de dos conductores, se separarán primero estos a lo largo de la distancia más apropiada para la posterior conexión y después se realizará sobre cada uno de ellos las operaciones descritas.

#### I.6.- TUBOS AISLANTES.

En muchas ocasiones se necesita añadir a los puntos de conexión de los cables y cablecillos una porción de tubo o cubierta con objeto de cubrir la zona soldada y aislarla eléctricamente de otros puntos próximos evitando cualquier riesgo de cortocircuito.

La porción de tubo necesaria, debe ser introducida en el cable previamente a la soldadura



desplazandose después hasta cubrir ésta, una vez -- que se halla enfriado por completo.

Este tubo aislante se conoce habitualmente con el nombre de "macarrón" y se adquiere en el comercio a la longitud que se desee.

Existe un tipo especial de tubo o "macarrón" denominado termorretráctil, que tiene la -- propiedad de contraer su diámetro al aplicarle calor. A pesar de su precio que es superior al de los demás tipos, tiene la ventaja de que una vez que se ha situado sobre la zona deseada se le puede calentar con el soldador u otros medios térmicos (evitando el contacto directo) con lo que quedará completamente comprimido y adaptado a la forma del punto de soldadura, obteniéndose una total inmovilidad, imposible de conseguir con otros "macarrones".

#### I.7.- MONTAJE DE DISIPADORES.

Existen algunos componentes y fundamentalmente semiconductores de potencia a los que es necesario incorporar un radiador o disipador de calor. Este radiador se situará sobre el componente, antes del montaje de éste sobre el circuito impreso empleando para ello un tornillo de sujeción. En los casos en que sea posible, deberá procurarse que el conjunto componente-radiador quede sujeto mecánicamente al circuito a través del mismo tornillo, conformando los terminales de una forma adecuada y añadiendo un separador de la altura suficiente para -- conseguir un aislamiento de la placa, debiendo estar realizado, por lo tanto, en un material que no --

transmita el calor.

Conviene señalar, por último, unos determinados modelos de transistores y tiristores de potencia cuyo montaje se realiza directamente sobre un radiador lo suficientemente grande para que sea capaz de evacuar el calor producido.

Para su instalación se emplean una lámina de mica que aísla el cuerpo del transistor del radiador y unas arandelas y tubos aislantes con los que se consigue el mismo efecto, sobre los tornillos de fijación y los terminales de salida. Normalmente estos conjuntos formados por el radiador y el semiconductor requiere un montaje separado del circuito impreso y se sitúan en una zona del equipo -- que disponga del máximo de ventilación para evacuar el calor generado.

La interconexión se realiza a través de un mazo de cablecillos hasta los puntos de enlaces que les correspondan. Tanto éste como el resto de mazos de cablecillos empleados en la construcción, deben de estar perfectamente fijados a la caja del aparato, con objeto de evitar cualquier riesgo de rotura o cortocircuito.

## APENDICE II.-

### DISEÑO DE CIRCUITOS IMPRESOS

Hoy día es casi impensable construir un equipo electrónico sin utilizar un circuito impreso por lo que puede considerársele como una herramienta más de trabajo. Pero ¿qué es un circuito impreso?

Se trata, simplemente, de los cables de interconexión entre los terminales de los componentes que forman el equipo electrónico. Tales "cables" tienen la particularidad de ser planos, en forma de cinta, y se encuentran "pegados", a una placa que les sirve de soporte. Habitualmente la misma placa sirve de soporte a los componentes.

Las cintas conductoras de un circuito impreso son de cobre, y se depositan sobre la placa-soporte por electrólisis, procedimiento químico que permite obtener capas muy finas del material sedimentado. El espesor, además, puede controlarse muy exactamente con tal proceso.

El grosor de las cintas de cobre que suele emplearse es de tan solo 35 micras (milésimas de milímetro), aunque en casos especiales pueden también encontrarse otros valores. Este débil espesor hace que las cintas presenten una sección (área transversal) muy pequeña. Por ejemplo, una cinta de 2 milímetros de ancho equivale a un cable de 0'3 milímetro de diámetro. A su vez, tan débil sección trae como consecuencia una relativamente alta resistencia eléctrica. Una cinta de 0'4 milí-

metros de ancho y 20 cm. de largo (que son corrientes en muchos circuitos modernos) tiene  $0'25\Omega$  de resistencia. Tales valores pueden llegar a ser limitativos en algunos casos.

Otra consecuencia de la pequeña sección es la poca capacidad para manejar corrientes elevadas que tienen tales conductores. Si por la cinta anterior de  $0'4$  milímetros de anchura, se hace circular una corriente de tan solo 100 mA, la densidad de corriente equivalente es de  $7 \text{ A/mm}^2$ , lo que es una cifra bastante elevada.

## II.2.- EL SOPORTE.

La placa que sirve de base a la cinta de cobre deben presentar algunas propiedades. La primera de todas es que debe ser aislante. Se tendrá en cuenta que el aislamiento debe conservarse en cualquier condición climática (temperatura, humedad) y, lo que es muy importante, tanto para corriente continua como para la frecuencia más elevada de señal que vaya a manejar el circuito.

Si, además, la placa va a servir de soporte mecánico a los distintos componentes del equipo debe presentar también la suficiente rigidez como para no ser deformada por aquellos. Otras condiciones, como el peso, la facilidad de tratamiento mecánico y otros factores similares sólo serán críticos en determinados casos.

Los dos tipos de material más ampliamente utilizados son la baquelita y la fibra de vidrio. La baquelita fue el primer material utiliza-

do cronológicamente hablando, sus características aislantes son buenas, aunque se degradan mucho con la frecuencia de trabajo del circuito. Además, la humedad puede también ocasionar pérdidas de aislamiento, por lo que no siempre es el material adecuado. Modernamente, el material que más se utiliza es, sin duda alguna, la fibra de vidrio, que presenta inmejorables características aislantes, incluso a las más elevadas frecuencias de funcionamiento, y en las más diversas condiciones climáticas.

La fibra de vidrio, además, presenta una mayor resistencia mecánica, y no tiende a resquebrajarse, como suele ocurrir con la baquelita.

Tan sólo presenta ésta frente a aquella una ventaja, y es su precio. Por esta razón la baquelita suele utilizarse en equipos que manejan señales de frecuencias bajas o corriente continua, o en aquellos casos en que el costo del conjunto sea un factor limitativo importante.

## II.2.- LAS PLACAS.

Los circuitos impresos no se fabrican "pegando" las distintas cintas que lo forman sobre el soporte y en el lugar adecuado, sino que siempre se parte de la llamada placa virgen, en la que sobre una de sus caras, o en las dos si el circuito va a ser de doble cara, está depositada una fina lámina "35 micras" de cobre electrolítico que la cubre por completo.

Sobre la placa virgen se dibuja (con

los materiales adecuados ,como veremos en su momento) el verdadero circuito de conexiones que se desee realizar. Todas aquellas zonas de la lámina de cobre que no hallan sido tapadas por el dibujo del circuito, se retiran del soporte empleando para ello un disolvente químico.

Por tanto, lo primero que debe confeccionarse es el circuito de conexiones que dessa materializarse. Para ello, es necesario partir del esquema del montaje a realizar, que es el "plano" de interconexión entre los distintos componentes empleados.

### II.3.- DIMENSIONES.

Para facilitar la labor de los fabricantes de equipos electrónicos (y de los aficionados), las dimensiones de la mayoría de los componentes están normalizadas, en particular las que se refieren a la separación entre los terminales de los mismos, que son las que más interesan a la hora del diseño del circuito impreso.

Tal normalización supone que la distancia entre los terminales adyacentes de cualquier componente es múltiplo de 0'1" (una décima de pulgadas), o lo que es lo mismo, de 2'54 milímetros. Las únicas excepciones las constituyen los elementos que no hayan de ir normalmente sobre circuito impreso (por ejemplo, transistores de potencia), algunos circuitos integrados con los terminales sobre plantilla circular o los componentes con terminales axiales que deben ser doblados a la me-

dida adecuada. Una vez en posesión del esquema de conexiones, lo más conveniente es hacerse con todos los componentes y obtener las dimensiones reales de cada uno de ellos. El tamaño físico de los mismos decidirá la separación mínima entre ellos, la distancia entre sus terminales, la ubicación de los taladros a realizar, etc.

Como regla práctica, puede codificarse la separación entre terminales como un número entero igual al número de veces que contienen la unidad normalizada (2'54 milímetros). Por ejemplo las patillas de un circuito integrado CD4011 están separadas una unidad (2'54 milímetros), mientras que la separación entre las dos filas de patillas es de tres unidades (7'62 milímetros). Los terminales de una resistencia de 1/3 ó 1/2 W pueden doblarse de forma que entre ellos haya una separación de cinco unidades (12'70 milímetros).

#### II.4.- EL DISEÑO.

No existen reglas fijas para el diseño del circuito de conexiones. Por lo general, se prefieren conexiones cortas y directas entre terminales. Existe una importante limitación y es que dos pistas de cobre no pueden cruzarse, como parece evidente. Tales casos suelen resolverse haciendo que una de las dos pista en conflicto dé un "rodeo", de forma que una de las dos pistas en conflicto dé un "rodeso", de forma que evite el cruce. Cuando no quede más solución que cruzar dos cintas, se recurre a hacer un "puente" con hilo de conexión por el lado de los componentes.

Lo primero que debe decidirse es si el diseño se va a hacer sobre placa de una o dos caras de cobre. Los circuitos con componentes discretos pueden ir sobre placa de una sola cara impresa. Los de alto grado de integración suelen requerir doble cara, salvo que se hagan gran cantidad de "puentes". La ventaja de la doble cara es que se resuelven gran cantidad de "cruces", llevando una cinta por cada cara. Por contra, el precio es algo mayor y se presentan problemas de interconexión entre ambas caras y de aislamiento de los componentes. En circuitos de una sola cara, los componentes se ubican en el lado opuesto al cobre, para evitar cortocircuitos o conexiones fortuitas inadecuadas. En circuitos de doble cara deben ubicarse en el lado que menor número de cintas contenga, como precaución suplementaria, es aconsejable dar una capa de barniz aislante antes de colocar los componentes. Es importante hacer el diseño pensando que los componentes se "ven" desde abajo en la cara opuesta a la de su ubicación.

Las cintas de cobre deben guardar una separación juiciosa que elimine la posibilidad de cortocircuito, resistencia de aislamiento baja o acoples indeseables que perturben el funcionamiento del circuito. Además, deben ser lo suficientemente anchas para permitir el paso de la corriente que por ellas vaya a circular.

Es conveniente ensanchar aquella zona de cinta que vaya a alojar el terminal del componente, se conseguirán mejores conexiones. Además,



debe evitarse que dos terminales lleguen al mismo taladro, lo correcto es hacer dos taladros unidos por la cinta correspondiente.

## II.5.- MECANIZADO DE PLACAS.

La placa de circuito impreso contiene los componentes y las interconexiones entre ellos necesarias para el funcionamiento de un circuito.

En la mayoría de los casos tal circuito necesita llevar también conexiones con el exterior del mismo, por ejemplo, la fuente de alimentación, la antena, un altavoz, el conmutador de funciones, un potenciómetro de regulación, etc.

Aunque los cables que lleven tales conexiones pueden soldarse directamente sobre los -- puntos adecuados ( y previstos para ello) del circuito impreso, tal sistema no resulta aconsejable - en la práctica, sino que lo más conveniente es la utilización de conectores especiales, ya sean de - canto, torretas, espadines, bayonetas u otros.

Con los conectores se evita hacer repetidas soldaduras y de soldaduras sobre las cintas impresas ( lo que podría llegar a levantarlas) además de permitir efectuar la soldadura (si el conector la necesita) por el lado de los componentes lo que resulta interesante en muchas ocasiones.

Otra forma de evitar que se levanten las cintas impresas ( incluso aquellas que no llevan soldaduras) es procurar que no presenten "es--quinas" en su trazado, haciendo que los recorridos en cambio de dirección lo sean en forma curva, más

bien que en forma angular más o menos pronunciada.

## II.6.- LOS TALADROS.

Una vez decidida la ubicación de los componentes sobre la placa y el trazado del circuito impreso, debemos obtener la placa del tamaño adecuado. En el mercado existen placas "virgenes" - de muy variadas dimensiones, así como del material base que deseemos, y tanto en una sola cara como - en doble cara.

Lo primero que debe hacerse es cortar la placa a la medida requerida por el diseño. Para ello, emplearemos preferiblemente una segueta equipada con una hoja de sierra para metal. No es aconsejable emplear hojas para madera, pues podrían estropearse, ni serruchos, pues deja un corte basto e impreciso, que posteriormente habría que retocar.

Con ayuda de una lima fina, o una lija, pueden eliminarse las pequeñas rebabas sobrantes.

Una vez la placa cortada, se procede a marcar los sitios en que deben ir los taladros.

Para ello situaremos una copia del diseño sobre la cara que va impresa. Con un buril, - un granete o, simplemente, un clavo de acero, se marcan todos los taladros. Atención a las placas de baquelita, que pueden levantarse al ser quebradizas o romperse con golpes fuertes. Si el circuito es de doble cara, el proceso anterior puede hacerse por cualquiera de las dos caras (aunque solamente por una de ellas). Solo habrá que tener en -

cuenta que la plantilla empleada corresponda a la cara por la que va a efectuarse los taladros. Si son necesarias varias placas iguales, pueden apilarse para efectuar todos los taladros de una vez.

Debe prestarse atención a que todas presenten la cara correcta y a que no puedan deslizarse unas sobre otras durante el proceso.

Con una taladradora adecuada (eléctrica o manual) se realizan todos los agujeros necesarios. El diámetro de los mismos dependen de los terminales de los componentes. Con una broca de 1 milímetro valdrá para la mayoría de los casos. Algunos de ellos, tales como espadines, potenciómetros y algunos otros, pueden necesitar una broca de 1'5 milímetros. Para los taladros que lleven tornillos, se empleará una broca de 3 milímetros.

## II.7.- TERMINACION MECANICA.

Las rebabas formadas al taladrar deben retirarse. Para ello no debe emplearse una lima o una lija, pues se rayaría todo el cobre, con el peligro de levantarlo o suprimirlo en algunas zonas. Debe pensarse que la película de cobre es de tan solo 35 milésimas de milímetro.

El mejor procedimiento consiste en emplear una broca de diámetro algo mayor que la empleada por el taladro, por ejemplo, una de 3 milímetros para taladros de 1 o 1'5 milímetros, o una de 6 milímetros para los de 3 milímetros. La broca se manejará con los dedos, girándola a un lado y otro hasta hacer desaparecer la rebaba. No emplear

taladradora para esta operación.

Antes de dar por terminada la mecanización es conveniente limpiar bien las placas de rebabas metálicas, virutas, polvo y otros materiales depositados sobre el cobre. Para ello puede emplearse un paño suave impregnado con alcohol o acetona, que eliminará restos de grasa o pegamento -- que pudieran quedar. Las placas ya mecanizadas y limpias deben tratarse como clichés fotográficos, evitando rayarlas o poner los dedos sobre la superficie cobreada.

## II.8.- TRANSCRIPCIÓN DEL CIRCUITO.

Una vez la placa de circuito impreso ha sido mecanizada, comienza la fase del tratado de la cara (o caras) de cobre, cuyo final es la obtención del circuito impreso propiamente dicha. Para formar el circuito debe eliminarse todo aquel cobre que no vaya a ser parte del mismo. La manera más usual de hacerlo es disolviéndolo de un agente químico.

El disolvente no sabe que cobre debe eliminar, por lo que debe protegerse contra la acción del mismo las partes que vayan a formar el -- circuito. Existen varias formas de hacerlo, entre ellas la protección con cintas y láminas adhesivas, con tintas especiales y utilizando placas especialmente tratadas.

## II.9.- CINTAS Y LÁMINAS.

Consiste este procedimiento en emple-

ar cintas de material plástico o de papel que lle-  
van adhesivos por una de sus caras y que pueden en-  
contrarse en diferentes anchos. Dicha cara se pega  
sobre el cobre en aquellos lugares que deben perma-  
necer para formar el circuito. En otros casos, so-  
bre todo para la formación de puntos de soldadura  
de los terminales de los componentes, en uniones -  
de cintas y otras situaciones similares suelen uti-  
lizarse "símbolos" preformados de los mismos, que -  
pueden depositarse en el lugar preciso por aplica-  
ción de la lámina de papel transparente que los --  
contiene, sobre el cobre.

Las ventajas del método son su rápi--  
dez y precisión en la formación del circuito impre-  
so, así como la pulcritud que presenta la placa --  
terminada. No obstante debe tenerse cierta pericia  
en el tratamiento de los materiales, que han de ser  
de buena calidad y no tiendan a despegarse. El ---  
principal inconveniente es el precio de las lámi--  
nas y cintas, compatibles únicamente con prototi--  
pos y circuitos originales que no puedan hacerse -  
por otro medio.

## II.10.- ROTULADORES.

Uno de los métodos más populares de -  
fabricación de circuitos impresos se basa en el em-  
pleo de rotuladores especiales, con tinta resisten-  
te al disolvente de cobre. Con tales rotuladores -  
se "pinta" el circuito deseado, incluido los punte-  
tos de soldadura y las uniones entre cintas.

Hay rotuladores de distintos gruesos de punta de forma que pueden emplearse para formar cintas de varios anchos o incluso para recubrir zonas grandes de cobre que actúen de masa o funcio--nes similares. La gran ventaja de este método es -su economía, aunque es necesario tener buen "pulsó" y ser un poco artista para que el diseño quede in--tachable. Debe procurarse emplear la suficiente de tinta para que el cobre quede bien recubierto, lo que puede hacerse pasando varias veces por el mis--mo punto el rotulador. Si no se hace así, a la ho--ra de disolver el cobre sobrante pueden quedar zo--nas con terminaciones poco presentables. En caso -de error la tinta puede borrarse con un trapo im--pregnado de acetona. El método también resulta ideal para prototipos y primeras pruebas.

## II.11.- FOTOGRAFADO.

Esta técnica resulta muy útil cuando se van a hacer gran cantidad de circuitos iguales, aunque puede emplearse también para un sólo ejem--plar. Para ello, se encuentran en el mercado pla--cas que lleva recubierto el cobre con un barniz fo--tosensible, como si se tratara de una película fo--tográfica. Son muy fáciles de reconocer, pues sue--len presentarse en bolsas o cajas negras y cerra--das, para no ser afectadas por la luz.

Se obtiene previamente un negativo --del circuito a realizar. El negativo no es impres--cindible que sea fotográfico, salvo que la calidad requerida así lo exija. Puede realizarse con papel

o película plástica transparente, sobre la que se dibujará dicho negativo.

Una vez realizado, el negativo se coloca sobre la cara de cobre tratada con barniz, - teneiendo la precaución de que se quede bien apoyado. El conjunto se expone a la acción de la luz, preferiblemente ultravioleta, tal como la que dan las lámparas de mercurio o los tubos fluorescentes del tipo "luz de día". También existen aparatos especialmente diseñados para este propósito. Con luz normal o luz solar directa el grabado requerirá de un mayor tiempo de exposición.

A continuación se procede a revelar - el barniz fotosensibilizado (igual que en fotografía), de manera que en aquellas partes iluminadas no se disolverá con la acción del líquido revelador, quedando protegido el cobre con el barniz en "positivo" igual que si se hubiera empleado cualquiera de los métodos anteriores.

Existen también en el mercado placas revestidas con barniz sobre el que ha de aplicarse en "positivo" del circuito impreso. En el proceso del revelado se disolverá aquel barniz que haya sido expuesto a la luz, quedando protegido el cobre que formará el circuito. Los materiales empleados en cada caso no son intercambiables entre sí, debiendo utilizar el específico para cada uno de ellos bien sea positivo o negativo.

El barniz que recubre las placas, tanto el positivo como el negativo, es poco sensible, por lo que pueden trabajarse con ellas durante el

tiempo suficiente para su preparación en condiciones de luz tenue.

## II.12.- ACABADO DEL CIRCUITO.

Una vez que la placa impresa tiene ya trazado el circuito sobre la cara de cobre (o sobre ambas si es doble cara), por uno u otro de los métodos de protección del mismo, se procede a realizar un ataque químico al mismo para eliminarle.

El disolvente más ampliamente utilizado es el cloruro férrico, que puede encontrarse en droguerías. Comercialmente, puede presentarse en pastillas para disolver en agua, o bien ya líquido con una concentración más o menos elevada.

Las pastillas se disuelven mejor en agua tibia. Con 100 gramos de cloruro férrico en pastillas disueltas en un cuarto de litro de agua habrá suficiente para atacar una placa de unos 100 cm<sup>2</sup>. Cuando se adquiere en forma líquida, suele venir bastante concentrado, por lo que es conveniente añadir agua en una proporción de una parte de cloruro férrico por una ó dos de agua.

Para el atacado del cobre puede utilizarse una cubeta, preferiblemente de material plástico, y que no vaya a contener posteriormente alimentos o líquidos. Las mejores son las de gran superficie y bordes bajos. El tamaño dependerá del de las placas que vayan a tratarse. Las placas se depositan sobre el fondo de la cubeta, y se cubren con la cantidad suficiente de cloruro férrico.



## II.13.- TIEMPO DEL PROCESO.

Según la concentración de cloruro férrico y la superficie de cobre a atacar, el proceso durará más o menos tiempo. Una placa de 100 cm.<sup>2</sup> de superficie ( una sola cara) con cloruro medianamente concentrado puede estar lista en unos 20 ó - 30 minutos. El tiempo puede acortarse si se mueve - el líquido sobre la placa, para que vaya renovándose.

Cuidado con las salpicaduras, pues -- producen manchas difíciles de quitar sobre superficies de madera o plásticas. Sobre la piel, dejan - un color amarillento que, con el tiempo, paciencia y muchos lavados, acaban por quitarse. No son peligrosas, aunque es un producto tóxico, y debe evitarse dejarlo al alcance de los niños tanto durante su utilización, como en su almacenamiento. Durante el proceso de atacado, puede sacarse de vez en cuando la placa, para observar como se va disolviendo el cobre. No emplear para ello materiales metálicos, pues se estropean con el cloruro. No debe apurarse demasiado el tiempo de disolución, pues podría levantarse la capa protectora del mismo, con lo que la terminación final sería de dudosa calidad y presentación.

El cloruro férrico puede emplearse varias veces consecutivamente para distintas placas, aunque va perdiendo eficacia y alargándose el tiempo necesario para la disolución, debido al proceso químico que ocurre en el mismo. El cloruro ya empleado no puede filtrarse ni admite realizar sobre él procesos similares para aumentar su concentración.

ón, dado que los fenómenos que intervienen, son como ya se han mencionados puramente químicos.

#### II.14.- ACABADO FINAL.

Una vez que ha sido eliminado todo el cobre sobrante, se procede a lavar bien la placa con agua tibia para eliminar los restos de cloruro férrico que pudieran subsistir. Una vez seca se elimina la protección del cobre que forma el circuito, bien retirando las tiras adhesivas, o disolviendo el barniz, lo que puede hacerse con acetona.

Con el cobre al descubierto, se hace una buena limpieza con alcohol o acetona, para eliminar todo resto del proceso de formación del circuito. Cuando el cobre quede limpio, es conveniente aplicar un barniz que evite su oxidación y permita hacer soldaduras perfectas. También puede procederse al estañado de las cintas, proceso ya explicado, y que además de actuar de protección, ayudará a realizar buenas soldaduras.

No debe omitirse una revisión ocular, con la que pueden detectarse cortocircuitos entre tiras próximas o cintas que puedan encontrarse cortadas. El primer problema puede resolverse con ayuda de una cuchilla de afeitar o alguna "herramienta" similar, el segundo problema suele encontrar solución con el mismo proceso de estañado.

En caso de duda en algún punto, puede comprobarse la calidad del circuito terminado con un hómetro. Si dos cintas en cortocircuito (cuando no deben estarlo), la resistencia entre será cero.

o de un valor muy bajo (cuando debía ser infinita, -  
o muy elevada). Por el contrario, un corte en una  
cinta (que no debería existir) se detecta compro-  
bando la resistencia entre extremos de la misma: -  
si el corte existe, la resistencia será muy eleva-  
da, si no existe (cinta correcta), la resistencia  
será nula.

## APENDICE III

### MONTAJE Y AJUSTE DEL ANALIZADOR

Nuestro circuito está constituido por varias placas de circuito impreso diferente: de base, del generador de ruido rosa (con energía constante por octava), de entrada, 4 de filtros y una de visualización.

Por supuesto, tal composición hace -- que sea un montaje grande y que necesite un ajuste por etapas.

La construcción de las placas han tenido que pasar por varias etapas sucesivas que explicaremos a continuación.

#### ...DISEÑO.-

Esta parte solo se puede acometer una vez que tengamos el circuito eléctrico y se disponga de todos los componentes necesarios. Es la más difícil en la fabricación de una placa de circuito impreso. Cualquier error trae consigo que todo el trabajo posterior no sirva para nada.

Si la placa va a albergar un circuito que en principio resulta sencillo, se intentará diseñarla mediante el trazado de las pistas por una cara sola. Este método es el más fiable y el más sencillo, y la utilización de puentes está más que justificada.

Sin embargo, existen casos en que la densidad de pista es tan grande, que es necesario el uso de placas a doble cara facilitando el dise-

ño del circuito, el uso de pistas horizontales por una cara y de verticales por la otra, pasando de -- una cara a la otra mediante un cilindro metálico - que se suelda a ambos lados.

En caso de excesiva complejidad, el - diseño de las placas se realiza mediante ordenador.

Las placas a doble cara necesitan que los zocalos paraintegrados puedan ser soldados por ambas caras, por lo que los zocalos convencionales no pueden emplearse si la placa no dispone de lo - que se conoce con el nombre de taladros metaliza-- dos, consistentes en pequeños cilindros metálicos que mediante proceso de electrolisis se deposita - en los agujeros de la placa, uniendo si es el caso la pista de la cara superior con la pista de la ca ra inferior. Así, al soldar los componentes, solo es necesario hacerlo por una cara, ya que el esta ño sube por capilaridad por el taladro metalizado hacia la otra.

El diseño de las placas se ha hecho - en papel milimetrado dibujando los componentes a - tamaño real, y teniendo cuidado de señalar bien -- los puntos correspondientes a los taladros por los que vayan a penetrar los terminales de estos para pasar de una cara a la otra. Despues se unen con - lapiz todos aquellos puntos entre los que deben -- existir una conexión eléctrica mediante trazos de 1'5 milímetros de ancho aproximado. En el caso de que circulen intensidades de corrientes elevadas - se debe aumentar la anchura de estos trazos lo que sucede para corrientes superiores a los 0'5 amperi os.

Los trazos se dibujan siguiendo líneas rectas, formando unos con otros ángulos de 45 y 90 grados.

En el dibujo se marcan las polaridades de todos aquellos componentes que únicamente admitan una posición de montaje, tales como condensadores electrolíticos, diodos, etc.

Durante el diseño es muy conveniente una serie de recomendaciones a tener en cuenta, -- que se pueden resumir en las siguientes:

.- La distancia mínima que se debe dejar entre dos puntos próximos, no unidos entre sí, será de 5 milímetros.

.- La separación entre terminales de los diversos componentes se medirá con un calibre o instrumento similar, antes de realizar su dibujo.

.- En las entradas y salidas del circuito impreso se emplearán terminales del tipo espadín, ya que resultan muy adecuados para la soldadura de cables.

.- Los taladros para los tornillos de sujeción de la placa serán de 3'5 a 4 milímetros y se dibujará a una distancia tal de las conexiones que se evite cualquier problema de cortocircuito entre ellas y los separadores metálicos de fijación.

Una vez que se ha completado el diseño sobre el papel milimetrado se trazarán los bordes de este, delimitando así la superficie que ocupará el circuito impreso.

### ...FABRICACION.-

El método seguido para la fabricación

de las placas ha sido el que se explicó en el apendice anterior con el nombre de Fotograbado.

El primer paso ha sido hacer una copia del papel milimetrado en papel vegetal, de donde se ha obtenido posteriormente una transparencia de cada una de las placas del analizador. Será con esa transparencia con la que se pasará al proceso de fotograbado, con el cual finalizaremos la fabricación de las placas procediendo a continuación al taladrado de las mismas.

### ...MONTAJE.-

Lo más importante en el montaje de -- los componentes es respetar las tolerancias sitadas y recomendamos también la utilización de zocalos de buena calidad para los circuitos integrados.

En los circuitos de entrada y alimentación contenidos en la misma placa, solo se emplea unas pocas resistencias del 1% de tolerancia. Su valor se indica mediante 4 bandas anulares de colores, en lugar de las 3 habituales. La función de las tres primeras es exactamente la misma que la que tienen las dos primeras de las resistencias ordinarias. La cuarta banda es la del multiplicador y se aplica en la forma acostumbrada. Por ejemplo, una resistencia ordinaria de 1K tiene los colores marrón, negro y rojo, la misma resistencia con una tolerancia del 1% tendrá: marrón, negro, negro y - marrón. La segunda banda negra indica que la tercera cifra es cero y la última banda marrón indica que hay que multiplicar el número obtenido con las

tres primeras bandas (100 en este caso) por 10.

Los reguladores de tensión integrados deben estar provistos de disipadores. Los conmutadores, diodos LED, etc., no deben conectarse hasta que el circuito completo haya alcanzado la etapa en la que puedan seleccionar la caja y esté hecha la disposición del panel frontal.

Sobre los filtros hay que decir que se han empleado cuatro placas de circuito impreso, tres de las cuales contienen ocho filtros cada una y la cuarta solo contiene seis. En estas placas todas las resistencias son del 1% de tolerancia. Idealmente los condensadores deberían ser del 2'5%, pero esta es una exigencia difícil de cumplir si queremos unas placas con unas dimensiones no demasiado grande. En efecto, más allá de unos 10nF, el tamaño de los condensadores Styroflex se hace inadmisibile. Para paliar este inconveniente hemos adoptado una solución de compromiso utilizando condensadores MKH o MKM del 5% para los valores superiores a 10nF. En la práctica, la precisión de los condensadores del 5% suele ser mejor que el 3%. Por debajo de los 10 nF el tamaño de los condensadores Styroflex no es excesivo. Si queremos mantener el analizador en un coste lo más moderado posible, todos los condensadores de filtro podrían ser tipo MK. La placa está diseñada para permitir esta posibilidad.

Todas las resistencias y condensadores de poliestireno están montados verticalmente en la placa. Para los circuitos integrados hay una



alternativa más barata. En las dos placas "más bajas", los circuitos integrados LM324 pueden utilizarse en lugar de los 741084. No ha de encontrarse ningún problema ya que ambos circuitos integrados tienen patillajes compatibles.

La placa de base contiene la sección de alimentación y los 30 rectificadores. Los dos reguladores de tensión deben estar provistos de radiador. Deberá instalarse terminales de soldadura en donde vayan a montarse las otras placas (con la excepción de la visualización), El cursor de cada potenciómetro ajustable (P1,...,P30) debe girarse ahora hasta su posición límite más cercana al diodo.

Todos los componentes, salvo los diodos LEDs y las resistencias R23...R52, pueden soldarse en su lugar sin más. Al acabar esta operación se podrán colocar estos diodos LEDs. Han de montarse en una hilera de 30 LEDs cada vez, teniendo cuidado al alinearlos. Finalmente, las resistencias R23...R52, van soldadas en la cara opuesta de la placa. Cada resistencia se suelda en sentido vertical y se conecta a la patilla del último LED de la hilera. El orificio practicado en el circuito para el interruptor S1 solo podrá utilizarse si este posee una palanca lo suficientemente larga -- como para alcanzar (y sobrepasar la cara frontal del panel).

El montaje del generador de ruido se simplifica enormemente gracias a la placa de circuito impreso que presentamos en este proyecto. Solo han de realizarse tres conexiones a la placa base:

+, - y 0. La tarea de efectuar estas conexiones se facilitará empleando terminales de soldadura en -- ambas placas. La soldadura de los terminales es -- mucho más fiable que la simple utilización de unos trozos de hilo. Los otros puntos de conexión se ne cesita para enlaces con los elementos de control y el bus de salida para la sección del ruido rosa.

El conmutador S1 es un pulsador de in versión.

El generador de ruido rosa se puede - utilizar como unidad autónoma. Todo lo que se re-- quiere es una alimentación simétrica de 8...12 V.

Como el consumo de corriente es muy - reducido bastarán dos pilas o una simple batería.

Cuando la totalidad de potenciómetro conmutadores, LEDs y conectores esten montados, po rán cablearse. Uno de los mejores tipos de conecto res que puede emplearse es el que suele utilizarse como conexión de "fono" en los amplificadores. La entrada se conecta a la placa de entrada con cable blindado. Este es el único lugar donde la masa del circuito puede conectarse a la caja. Ha de emplear se una conexión hembra de estéreo de 6'3mm. para - la entrada microfónica. Para fines de medida se su ele utilizar un micrófono electret( de electreto).

Una lámina de vidrio acrílico puede colocarse encima de la ventana del visualizador -- (el color más conveniente sería el rojo oscuro), en este caso la placa base debe ajustarse en la pñaga base no solo para que los LEDs coincidan con la es cala del panel frontal, sino para que estén lo más cerca posible de la lámina.

Antes de conectar el analizador es -- conveniente comprobar las conexiones al transformador. Los terminales de 10 V. deben conectarse a los puntos de suministro de la tensión de c.a. en la placa base y los de 15 V. han de unirse a los puntos de c.a. en la placa de entrada. El terminal de tierra del transformador ha de conectarse al punto correspondiente en ambas placas.

### ... MICROFONOS Y MEDIDAS.-

El analizador de audio es bastante -- sencillo de manejar. El ruido rosa se aplica a la entrada del equipo que ha de analizarse y la salida de éste se conecta a la entrada del analizador.

Cuando estén adecuadamente ajustados el conmutador de escalas y el potenciómetros de niveles, aparecerá en la pantalla la característica de la frecuencia. A continuación se podrá utilizar el conmutador de resolución para establecer el margen de visualización. Al gunas de las aplicaciones más comunes son:

.- Medida de la característica de frecuencia de un amplificador.

Esta curva suele carecer de interés -- pero puede servir, por ejemplo, para comprobar el efecto de un corrector de la tonalidad.

.- Medida de característica de frecuencia de una grabadora de cinta.

En este caso el ruido rosa ha de grabarse a un nivel bajo, tal como -20 ó -30 VU, para evitar que se produzca distorsión.

.- Estudio de la acústica de un recinto.

Se trata de una aplicación especialmente importante del analizador. En este caso se utiliza un micrófono de medida. Trabajando con un ecualizador de  $1/3$  de octava y el analizador, la cacterística de frecuencia en la zona de escucha - puede hacerse casi completamente plana. El analizador y un micrófono de medida proporcionan un método ideal para examinar un altavoz. Cualquiera que sea la clase de medida efectuada es preciso pres--tar la máxima atención al nivel de la señal de ruido inyectada en el aparato de prueba con el fin de no saturarlo. Los picos en las bandas de frecuencias más bajas son unos 10dB más elevados que a 1kHz.

En estas bandas, la visualización nunca proporcionará una salida perfectamente estable, sino más bien algo saltarina, lo que se debe al hecho de que el ancho de banda de los filtros es proporcionalmente más estrecha con lo que muy pocas - señales aparecerán en una banda con cualquier duración dada.

Finalmente, algunas observaciones sobre el micrófono de medida, parte esencial del analizador. Debe tener una buena característica de frecuencia y no ser demasiado caro. Una posibilidad adecuada sería la cápsula KE 4-211-2 de Sennheiser, - que tiene un tamaño similar al de un BC 547 y cuya cacterística de frecuencia es rectilínea entre - 40 Hz y 20 kHz, con una desviación no superior a -  $\pm 2.5$  dB. Se trata de un micrófono de electreto -- provisto de un FET tampón que hay que alimentar --

con un cable distinto al de la señal (+5 V). Esta tensión podrá obtenerse a partir de la alimentación estabilizada de +12V en el circuito de entrada o en el de base. Hay muchos otros modelos de micrófonos propuestos por tantos otros fabricantes, se dispone de opciones bastante amplia. El nivel del sonido - medido puede ajustarse por medio de P1 (y, si fuera necesario, cambiando R2) en la placa de entrada. Este ajuste no puede efectuarse realmente sin un medidor de la presión acústica, pero solo lo precisará si quiere medir valores absolutos.

Para realizar medidas a partir de una referencia absoluta tendremos que calibrar el analizador en tiempo real con la ayuda de otro analizador ya calibrado.

**Applications**

**8 bit D to A converter**

The ZN425E gives an analogue voltage output directly from pin 14 therefore the usual current to voltage converting amplifier is not required. The output voltage drift, due to the temperature coefficient of the Analogue Output Resistance R will be less than 0.004% per °C (or 1 L.S.B./100°C) if R is chosen to be  $\geq 650k\Omega$ .

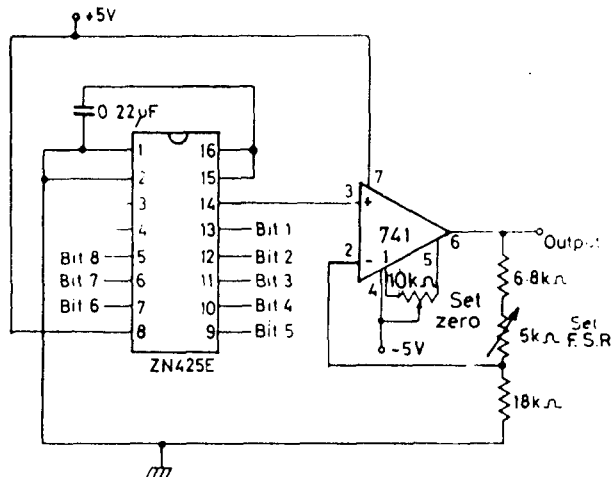
In order to remove the offset voltage and to calibrate the converter a buffer amplifier is necessary. Fig 2 shows a typical scheme using the internal reference voltage. To minimise temperature drift in this and similar applications the source resistance to the inverting input of the operational amplifier should be approximately  $6k\Omega$ .

The calibration procedure is as follows:

- i. Set all bits to OFF (low) and adjust  $R_2$  until  $V_{out} = 0.000V$ .
- ii. Set all bits to ON (high) and adjust  $R_1$  until  $V_{out} =$  Nominal full scale reading  $- 1$  L.S.B.
- iii. Repeat i. and ii.

e.g. Set F.S.R. to  $+3.840$  volts  $- 1$  L.S.B.  
 $= 3.825$  volts  
 $\frac{3.84}{256} = 15.0$  millivolts.)

Figure 2: 8 bit digital to analogue converter



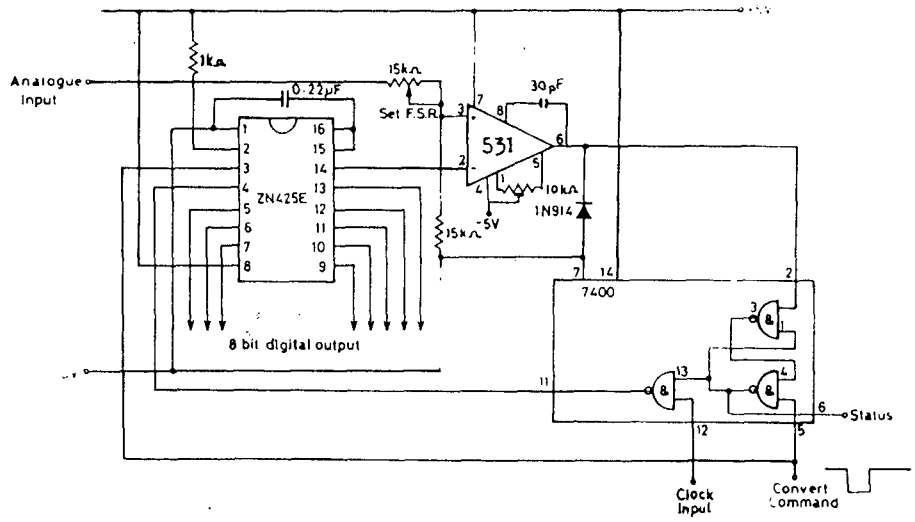
**8 bit analogue to digital converter**

A counter type analogue to digital converter can be constructed by adding a voltage comparator as in Fig 3. On the negative edge of the CONVERT COMMAND Pulse the counter is set to zero and the STATUS output to logical 1. On the positive edge the counter starts to count up from zero. The analogue output ramps until it equals the analogue voltage applied to the other input of the comparator. At this point, any further clock pulses are inhibited and STATUS goes low to indicate that the output data is valid.

The conversion time depends upon the value of the analogue input and for full scale reading is given by the clock frequency divided into the maximum number of counts.

For example if  $F_{clock} = 256K$   
 conversion for F.S.R.) =  $\frac{2^8}{256,000}$  seconds  
 $= 1$  milliseconds

Figure 3: 8 bit analogue to digital converter



ELECTRICAL CHARACTERISTICS (continued)

Parameter	Test conditions	Values						Unit			
		V <sub>DD</sub> (V)	T <sub>Low</sub> *		25°C				T <sub>High</sub> *		
			Min.	Max.	Min.	Typ.	Max.		Min.	Max.	
<b>CONTROL (V<sub>C</sub>)</b>											
V <sub>ILC</sub> Control input Low voltage	I <sub>is</sub>   < 10 μA V <sub>is</sub> = V <sub>SS</sub> , V <sub>os</sub> = V <sub>DD</sub> and V <sub>is</sub> = V <sub>DD</sub> , V <sub>os</sub> = V <sub>SS</sub>	5		1			1		1	V	
		10		2			2		2		
		15		2			2		2		
V <sub>IHC</sub> Control input High voltage		5	3.5		3.5			3.5		V	
		10	7		7			7			
		15	11		11			11			
I <sub>IH</sub> , I <sub>IL</sub> Input leakage current	HCC types	V <sub>is</sub> < V <sub>DD</sub> V <sub>DD</sub> -V <sub>SS</sub> = 18V	18		±0.1		±10 <sup>-5</sup>	±0.1		±1	μA
	HCF types	V <sub>DD</sub> -V <sub>SS</sub> = 15V V <sub>CC</sub> < V <sub>DD</sub> -V <sub>SS</sub>	15		±0.3		±10 <sup>-5</sup>	±0.3		±1	
Crosstalk (control input to signal output)	V <sub>C</sub> = 10V (Sq. wave) t <sub>r</sub> , t <sub>f</sub> = 20 ns R <sub>L</sub> = 10 kΩ	10				50				mW	
Turn-On propagation delay	V <sub>IN</sub> = V <sub>DD</sub> t <sub>r</sub> , t <sub>f</sub> = 20 ns C <sub>L</sub> = 50 pF R <sub>L</sub> = 1 kΩ	5				35	70			ns	
		10				20	40				
		15				15	30				
Control input Repetition rate	V <sub>is</sub> = V <sub>DD</sub> , V <sub>SS</sub> = GND R <sub>L</sub> = 1 kΩ to gnd C <sub>L</sub> = 50 pF V <sub>C</sub> = 10V (Square wave centered on 5V) t <sub>r</sub> , t <sub>f</sub> = 20 ns V <sub>os</sub> = ½ V <sub>os</sub> @ 1KHz	5				6				MHz	
		10				9					
		15				9.5					
C <sub>I</sub> Input capacitance	Any input					5	7.5			pF	

- \* T<sub>Low</sub> = - 55°C for HCC device; -40°C for HCF device.
- \* T<sub>High</sub> = +125°C for HCC device; +85°C for HCF device.
- The Noise Margin for both "1" and "0" level is: 1V min. with V<sub>DD</sub> = 5V  
2V min. with V<sub>DD</sub> = 10V  
2.5V min. with V<sub>DD</sub> = 15V

ANALOG MULTIPLEXERS/DEMULTIPLEXERS:

4067B SINGLE 16-CHANNEL  
4097B DIFFERENTIAL 8-CHANNEL

- LOW ON RESISTANCE: 125Ω (TYP.) OVER 15 V<sub>p-p</sub> SIGNAL-INPUT RANGE FOR V<sub>DD</sub>-V<sub>SS</sub> = 15V
- HIGH OFF RESISTANCE: CHANNEL LEAKAGE OF ± 10 pA (TYP.) @ V<sub>DD</sub>-V<sub>SS</sub> = 10V
- MATCHED SWITCH CHARACTERISTICS: ΔR<sub>ON</sub> = 5Ω (TYP.) FOR V<sub>DD</sub>-V<sub>SS</sub> = 15V
- VERY LOW QUIESCENT POWER DISSIPATION UNDER ALL DIGITAL-CONTROL INPUT AND SUPPLY CONDITIONS: 0.2 μW (TYP.) @ V<sub>DD</sub>-V<sub>SS</sub> = 10V
- BINARY ADDRESS DECODING ON CHIP
- QUIESCENT CURRENT SPECIFIED TO 20V FOR HCC DEVICE
- STANDARDIZED SYMMETRICAL OUTPUT CHARACTERISTICS
- 5V, 10V, AND 15V PARAMETRIC RATINGS
- INPUT CURRENT OF 100 nA AT 18V AND 25°C FOR HCC DEVICE
- 100% TESTED FOR QUIESCENT CURRENT
- MEETS ALL REQUIREMENTS OF JEDEC TENTATIVE STANDARD No. 13A, "STANDARD SPECIFICATIONS FOR DESCRIPTION OF "B" SERIES CMOS DEVICES"

The HCC 4067B, HCC 4097B (extended temperature range) and HCF 4067B, HCF 4097B (intermediate temperature range) are monolithic integrated circuits, available in 24-lead dual in-line plastic or ceramic package and ceramic flat package.

The HCC/HCF 4067 and HCC/HCF 4097 COS/MOS analog multiplexers/demultiplexers are digitally controlled analog switches having low ON impedance, low OFF leakage current, and internal address decoding. In addition, the ON resistance is relatively constant over the full input-signal range.

The HCC/HCF 4067 is a 16-channel multiplexer with four binary control inputs, A, B, C, D, and an inhibit input, arranged so that any combination of the inputs selects one switch.

The HCC/HCF 4097 is a differential 8-channel multiplexer having three binary control inputs A, B, C, and an inhibit input. The inputs permit selection of one of eight pairs of switches. A logic "1" present at the inhibit input turns all channels off.

ABSOLUTE MAXIMUM RATINGS

V <sub>DD</sub> *	Supply voltage: HCC types HCF types	-0.5 to 20 -0.5 to 18	V V
V <sub>I</sub>	Input voltage	-0.5 to V <sub>DD</sub> + 0.5	V
I <sub>I</sub>	DC input current (any one input)	± 10	mA
P <sub>tot</sub>	Total power dissipation (per package) Dissipation per output transistor for T <sub>op</sub> = full package-temperature range	200	mW mW
T <sub>op</sub>	Operating temperature: HCC types HCF types	-55 to 125 -40 to 85	°C °C
T <sub>stg</sub>	Storage temperature	-65 to 150	°C

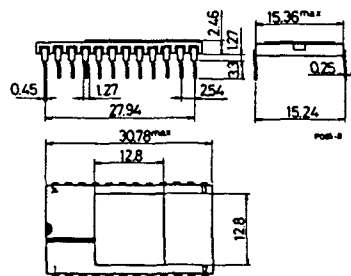
\* All voltage values are referred to V<sub>SS</sub> pin voltage

ORDERING NUMBERS:

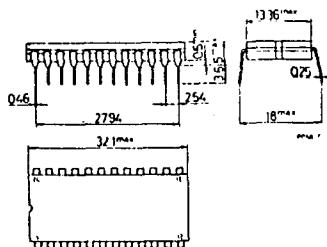
- HCC 4XXX BD for dual in-line ceramic package
- HCC 4XXX BF for dual in-line ceramic package, frit seal
- HCC 4XXX BK for ceramic flat package
- HCF 4XXX BF for dual in-line ceramic package, frit seal
- HCF 4XXX BE for dual in-line plastic package

MECHANICAL DATA (dimensions in mm)

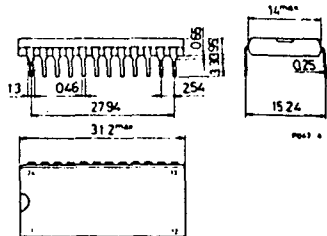
Dual in-line ceramic package for HCC 4XXX BD



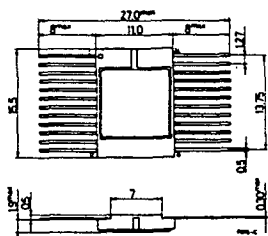
Dual in-line ceramic frit-seal package for HCC/HCF 4XXX BF



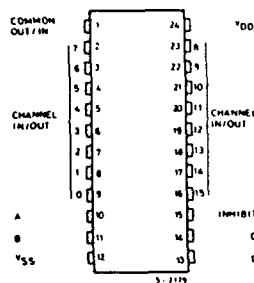
Dual in-line plastic package for HCF 4XXX BE



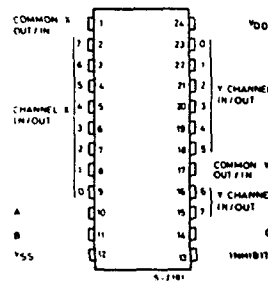
Ceramic flat package for HCC 4XXX BK



CONNECTION DIAGRAMS  
For HCC/HCF 4067B

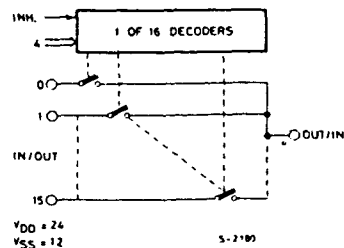


For HCC/HCF 4097B

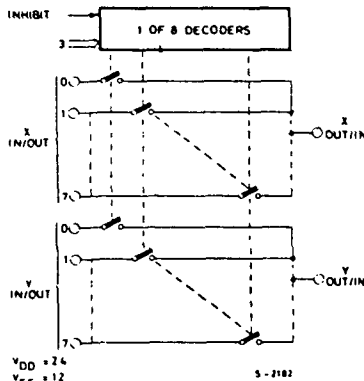


FUNCTIONAL DIAGRAMS

For HCC/HCF 4067B

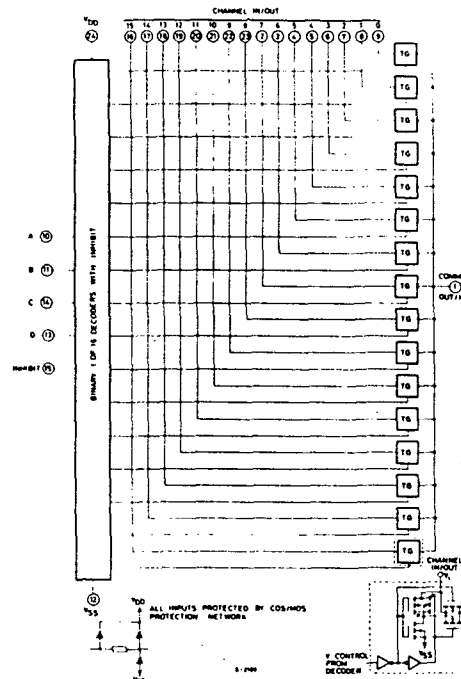


For HCC/HCF 4097B

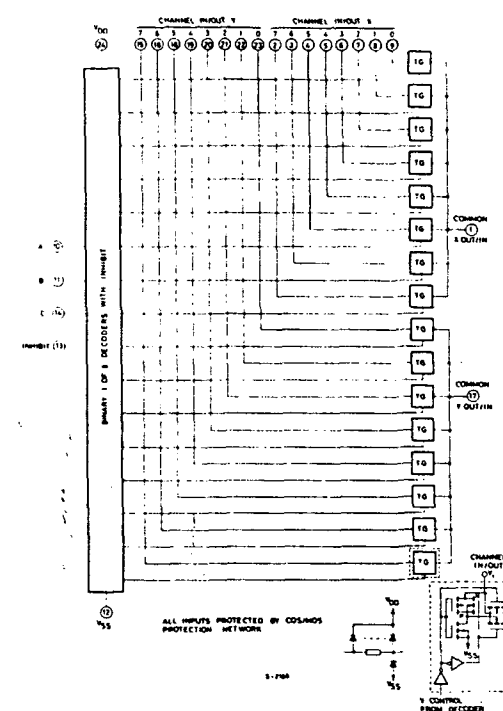


LOGIC DIAGRAMS

For HCC/HCF 4067B



For HCC/HCF 4097B



TRUTH TABLES

For HCC/HCF 4067B

A	B	C	D	Inh	Selected Channel
X	X	X	X	1	None
0	0	0	0	0	0
1	0	0	0	0	1
0	1	0	0	0	2
1	1	0	0	0	3
0	0	1	0	0	4
1	0	1	0	0	5
0	1	1	0	0	6
1	1	1	0	0	7
0	0	0	1	0	8
1	0	0	1	0	9
0	1	0	1	0	10
1	1	0	1	0	11
0	0	1	1	0	12
1	0	1	1	0	13
0	1	1	1	0	14
1	1	1	1	0	15

For HCC/HCF 4097B

A	B	C	Inh	Selected Channel
X	X	X	1	None
0	0	0	0	0X, 0Y
1	0	0	0	1X, 1Y
0	1	0	0	2X, 2Y
1	1	0	0	3X, 3Y
0	0	1	0	4X, 4Y
1	0	1	0	5X, 5Y
0	1	1	0	6X, 6Y
1	1	1	0	7X, 7Y

RECOMMENDED OPERATING CONDITIONS

V <sub>DD</sub>	Supply voltage: HCC types	3 to 18	V
	HCF types	3 to 15	V
V <sub>I</sub>	Input voltage	0 to V <sub>DD</sub>	V
T <sub>op</sub>	Operating temperature: HCC types	-55 to 125	°C
	HCF types	-40 to 85	°C



STATIC ELECTRICAL CHARACTERISTICS (over recommended operating conditions)

Parameter	Test conditions				Values						Unit		
	V <sub>IS</sub> (V)	V <sub>EE</sub> (V)	V <sub>SS</sub> (V)	V <sub>DD</sub> (V)	T <sub>Low</sub> (*)		25°C			T <sub>High</sub> (*)			
					Min.	Max.	Min.	Typ.	Max.	Min.		Max.	
I <sub>L</sub> Quiescent device current	HCC types				5	5	0.04	5	150	μA			
					10	10	0.04	10	300				
					15	20	0.04	20	600				
					20	100	0.08	100	3000				
					5	20	0.04	20	150				
					10	40	0.04	40	300				
HCF types					15	80	0.04	80	600				
									5	20	0.04	20	150
									10	40	0.04	40	300
									15	80	0.04	80	600
<b>SWITCH</b>													
ON Resistance	HCC types	0 < V <sub>I</sub> < V <sub>DD</sub>	0	0	5	800	470	1050	1300	Ω			
					10	310	180	400	580				
					15	200	125	240	320				
					HCF types	0 < V <sub>I</sub> < V <sub>DD</sub>	0	0	5		850	470	1050
					10	330	180	400	520				
					15	210	125	240	300				
					ΔON Resistance ΔR <sub>ON</sub> (Between any 2 channels)		0	0	5		10		
OFF(*) Channel Leakage Current	Any channel OFF	HCC types	0	0	18	100	±0.1	100	1000	nA			
	All channels OFF (common OUT/IN)	HCC types	0	0	18	100	±0.1	100	1000	nA			
	Any channel OFF	HCF types	0	0	15	300	±0.1	300	1000	nA			
	All channels OFF (common OUT/IN)	HCF types	0	0	15	300	±0.1	300	1000	nA			
C Capacitance	Input						5			pF			
	Output 4067						55						
	Output 4097			-5	5		35						
	Feedthrough						0.2						
<b>CONTROL (Address or Inhibit)</b>													
V <sub>IL</sub> Input low voltage		=V <sub>DD</sub> thru 1KΩ	V <sub>EE</sub> =V <sub>SS</sub> R <sub>L</sub> =1KΩ to V <sub>SS</sub> I <sub>IS</sub> <2μA (on all OFF channels)	5	1.5		1.5	1.5	1.5	V			
				10	3		3	3	3				
				15	4		4	4	4				
				V <sub>IH</sub> Input high voltage			5	3.5	3.5		3.5	V	
				10	7	7		7					
				15	11	11		11					
I <sub>IH</sub> , I <sub>IL</sub> Input leakage current	HCC types	V <sub>I</sub> = 0/18V		18	0.1	10 <sup>-3</sup>	0.1	1	μA				
	HCF types	V <sub>I</sub> = 0/15V		15	0.3	10 <sup>-3</sup>	0.3	1					
				C <sub>I</sub> Input capacitance	Any address or inhibit input			5	7.5	pF			

(\*) Determined by minimum feasible leakage measurement for automatic testing.

(\*) T<sub>Low</sub> = -55°C for HCC device; -40°C for HCF device.

T<sub>High</sub> = +125°C for HCC device; +85°C for HCF device.

DYNAMIC ELECTRICAL CHARACTERISTICS (T<sub>amb</sub> = 25°C, C<sub>L</sub> = 50 pF all input square wave rise and fall time = 20 ns)

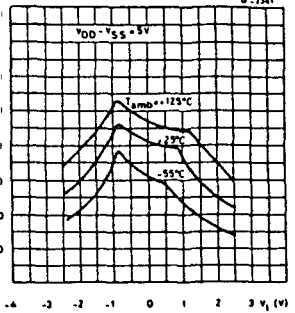
Parameter	Test conditions						Values		Unit	
	V <sub>C</sub> (V)	R <sub>L</sub> (kΩ)	f <sub>i</sub> (KHz)	V <sub>I</sub> (V)	V <sub>SS</sub> (V)	V <sub>DD</sub> (V)	Typ.	Max.		
<b>SWITCH</b>										
t <sub>pd</sub> Propagation delay time (Signal Input to output)	= V <sub>DD</sub>	200		5	0	5	30	60	ns	
						10	15	30		
						15	11	20		
Frequency Response Channel "ON" (Sine Wave Input) at 20 Log $\frac{V_o}{V_i} = -3dB$	= V <sub>DD</sub>	1		5(*)	0	10	V <sub>O</sub> at Common OUT/IN	4067B 14	MHz	
							V <sub>O</sub> at Any Channel	60		
Feedthrough (All channels OFF) at 20 Log $\frac{V_o}{V_i} = -40 dB$	= V <sub>SS</sub>	1		5(*)	0	10	V <sub>O</sub> at Common OUT/IN	4067B 20	MHz	
							V <sub>O</sub> at Any channel	8		
Frequency Signal Crosstalk at 20 Log $\frac{V_o(B)}{V_i(A)} = -40dB$	V <sub>C(A)</sub> =V <sub>DD</sub>	1		5(*)	0	10	Between Any 2 (A and B) channels		1	MHz
	V <sub>C(B)</sub> =V <sub>SS</sub>						Between sections (A and B) 4097B only		Measured on common	
		Measured on Any channel		18						
Sine wave Distortion f <sub>is</sub> = 1KHz sine wave	5	10	1	2(*)	0	5			0.3	%
	10	10	1	3(*)	0	10			0.2	
	15	10	1	5(*)	0	15			0.12	
<b>CONTROL (Address or Inhibit)</b>										
Propagation delay time: Address or Inhibit to signal OUT (channel turning ON)		10				0	5	325	650	ns
						0	10	135	270	
						0	15	95	190	
Propagation delay time: Address or Inhibit to signal OUT (channel turning OFF)		0.3				0	5	220	440	ns
						0	10	90	180	
						0	15	65	130	
Address or Inhibit to Signal Crosstalk		10*				0	10	75		mV peak

(\*) Peak to peak voltage symmetrical about  $\frac{V_{DD} - V_{SS}}{2}$

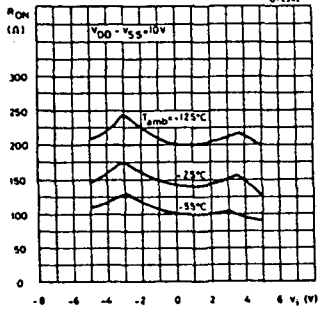
(\*) Both ends of channel



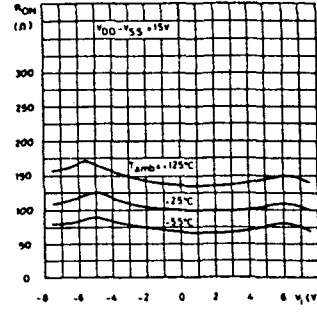
Typical ON resistance vs. input signal voltage (all types)



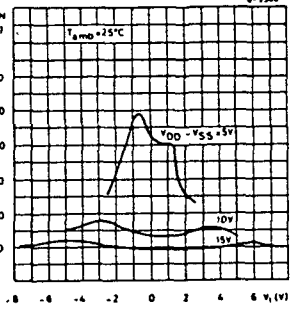
Typical ON resistance vs. input signal voltage (all types)



Typical ON resistance vs. input signal voltage (all types)

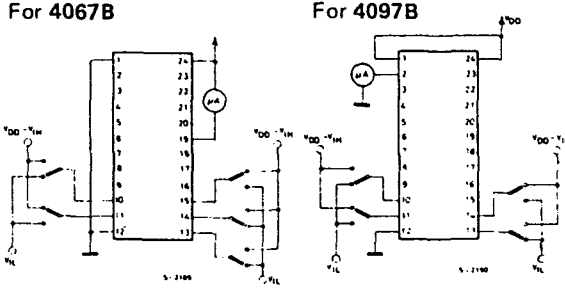


Typical ON resistance vs. input signal voltage (all types)

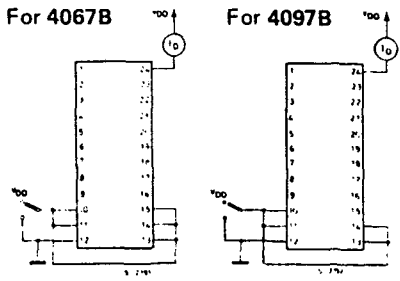


TEST CIRCUITS (continued)

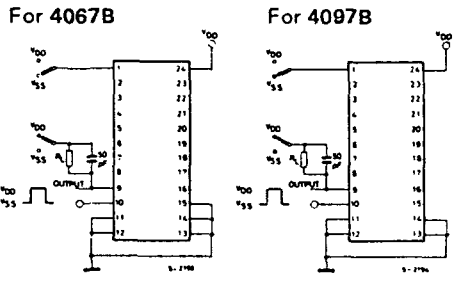
Input voltage-measure < 2 μA on all OFF channels (e.g. channel 12)



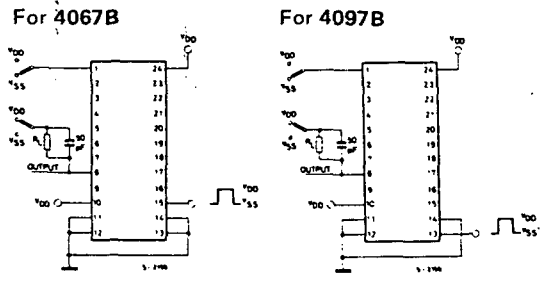
Quiescent device current



Turn-on and turn-off propagation delay-address select input to signal output (e.g. measured on channel 0)

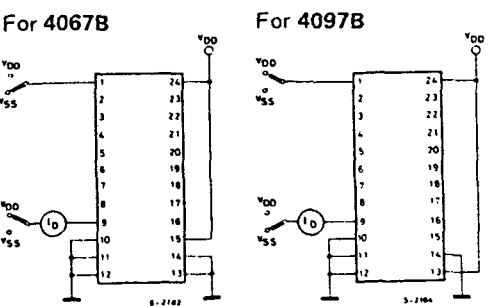


Turn-on and turn-off propagation delay-inhibit input to signal output (e.g. measured on channel 1)

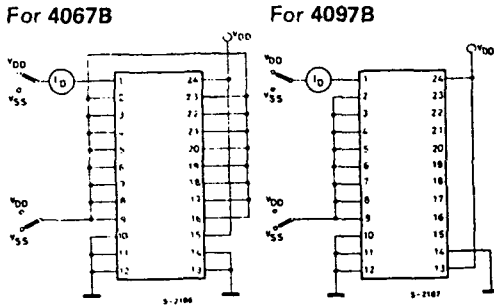


TEST CIRCUITS

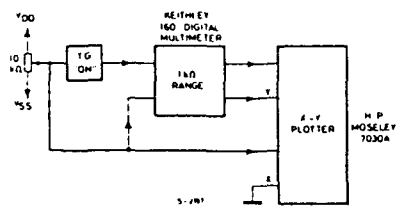
OFF channel leakage current-any channel OFF



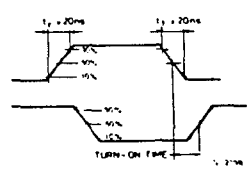
OFF channel leakage current-all channels OFF



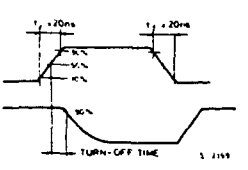
Channel ON resistance measurement circuit



Propagation delay waveform channel being turned ON (RL = 10 kΩ, CL = 50 pF)



Propagation delay waveform channel being turned OFF (RL = 300Ω, CL = 50 pF)





### APPLICATIONS INFORMATION

In applications where separate power sources are used to drive  $V_{DD}$  and the signal inputs, the  $V_{DD}$  current capability should exceed  $V_{DD}/R_L$  ( $R_L$  = effective external load). This provision avoids permanent current flow or clamp action on the  $V_{DD}$  supply when power is applied or removed from the HCC/HCF 4067B or HCC/HCF 4097B.

When switching from one address to another, some of the ON periods of the channels of the multiplexers will overlap momentarily, which may be objectionable in certain applications. Also when a channel is turned on or off by an address input, there is a momentary conductive path from the channel to  $V_{SS}$ , which will dump some charge from any capacitor connected to the input or output of the channel. The inhibit input turning on a channel will similarly dump some charge to  $V_{SS}$ .

The amount of charge dumped is mostly a function of the signal level above  $V_{SS}$ . Typically, at  $V_{DD} - V_{SS} = 10V$ , a 100 pF capacitor connected to the input or output of the channel will lose 3-4% of its voltage at the moment the channel turns on or off. This loss of voltage is essentially independent of the address or inhibit signal transition time, if the transition time is less than 1-2  $\mu s$ . When the inhibit signal turns a channel off, there is no charge dumping to  $V_{SS}$ . Rather, there is a slight rise in the channel voltage level (65 mV typ.) due to capacitive coupling from inhibit input to channel input or output. Address inputs also couple some voltage steps onto the channel signal levels.

In certain applications, the external load-resistor current may include both  $V_{DD}$  and signal-line components. To avoid drawing  $V_{DD}$  current when switch current flows into the transmission gate inputs, the voltage drop across the bidirectional switch must not exceed 0.8 volt (calculated from  $R_{ON}$  values shown in ELECTRICAL CHARACTERISTICS CHART). No  $V_{DD}$  current will flow through  $R_L$  if the switch current flows into terminal 1 on the HCC/HCF 4067B, terminals 1 and 17 on the HCC/HCF 4097B.

# COS/MOS INTEGRATED CIRCUIT



### 8-INPUT NAND/AND GATE

- MEDIUM-SPEED OPERATION -  $t_{PHL}$ ,  $t_{PLH} = 75$  ns (TYP.) AT 10V
- BUFFERED OUTPUT
- QUIESCENT CURRENT SPECIFIED TO 20V FOR HCC DEVICE
- 5V, 10V, AND 15V PARAMETRIC RATINGS
- INPUT CURRENT OF 100 nA AT 18V AND 25°C FOR HCC DEVICE
- 100% TESTED FOR QUIESCENT CURRENT
- MEETS ALL REQUIREMENTS OF JEDEC TENTATIVE STANDARD No. 13A, "STANDARD SPECIFICATIONS FOR DESCRIPTION OF "B" SERIES CMOS DEVICES"

The HCC 4068B (extended temperature range) and HCF 4068B (intermediate temperature range) are monolithic integrated circuit, available in 14-lead dual in-line plastic or cerami package, ceramic flat package and plastic micropackage. The HCC/HCF 4068B NAND/AND gate provides the system designer with direct implementation of the positive-logic 8-input NAND and AND functions and supplements the existing family of COS/MOS gates.

### ABSOLUTE MAXIMUM RATINGS

$V_{DD}^*$	Supply voltage: HCC types HCF types.	-0.5 to 20 -0.5 to 18	V V
$V_i$	Input voltage	-0.5 to $V_{DD}$ +0.5	V
$I_i$	DC input current (any one input)	$\pm 10$	mA
$P_{tot}$	Total power dissipation (per package) Dissipation per output transistor for $T_{op}$ = full package-temperature range	200 100	mW mW
$T_{op}$	Operating temperature: HCC types HCF types	-55 to 125 -40 to 85	°C °C
$T_{stg}$	Storage temperature	-65 to 150	°C

\* All voltage values are referred to  $V_{SS}$  pin voltage

### ORDERING NUMBERS:

- HCC 4068 BD for dual in-line ceramic package
- HCC 4068 BF for dual in-line ceramic package, frit seal
- HCC 4068 BK for ceramic flat package
- HCF 4068 BE for dual in-line plastic package
- HCF 4068 BF for dual in-line ceramic package, frit seal
- HCF 4068 BM for plastic micropackage

## 2. I/O PORT ALLOCATION

Utilization	Port No.	Application
RS-232-C	&H80	Data read-out/write-in
	&H81	Mode set (during write-in) Status (during read-out)
Printer	&H90	Strobe (during write-in) Status (during read-out)
	&H91	Data write-in
VDP	&H98	Data read-out/write-in with video RAM:
	&H99	Command, address set (during write-in) Status (during read-in)
PSG	&HA0	Address latch (write-in)
	&HA1	Data write-in
	&HA2	Data write-out
PPI	&HA8	Data read-out/write-in for port A (Memory slot select) use.
	&HA9	Data read-out/write-in for port B (key board scan) use.
	&HAA	Data read-out/write-in for port C (cassette).
	&HAB	Mode set (write-in)

- I/O addresses from &H00 to &H7F are not used. Addresses other than the above addresses of the address among &H80 to &HFF are reserved for system use.

## 3. CHARACTERS

### CHARACTERS HANDLED BY MXS-BASIC

The characters shown in the following character code table can be displayed.

Hexa- decimal code	00 - 1F		20 - 3F		40 - 5F		60 - 7F	
	code	character	code	character	code	character	code	character
0	0	(null)	32	(space)	64	@	96	'
1	1	☺	33	!	65	A	97	a
2	2	☹	34	"	66	B	98	b
3	3	♥	35	#	67	C	99	c
4	4	♦	36	\$	68	D	100	d
5	5	♣	37	%	69	E	101	e
6	6	♠	38	&	70	F	102	f
7	7	.	39	'	71	G	103	g
8	8	◼	40	(	72	H	104	h
9	9	○	41	)	73	I	105	i
A	10	◉	42	*	74	J	106	j
B	11	♂	43	+	75	K	107	k
C	12	♀	44	,	76	L	108	l
D	13	♪	45	-	77	M	109	m
E	14	♫	46	.	78	N	110	n
F	15	♫	47	/	79	O	111	o
0	16	+	48	0	80	P	112	p
1	17	⊥	49	1	81	Q	113	q
2	18	⊥	50	2	82	R	114	r
3	19	⊥	51	3	83	S	115	s
4	20	⊥	52	4	84	T	116	t
5	21	⊥	53	5	85	U	117	u
6	22		54	6	86	V	118	v
7	23	—	55	7	87	W	119	w
8	24	┌	56	8	88	X	120	x
9	25	└	57	9	89	Y	121	y
A	26	└	58	:	90	Z	122	z
B	27	└	59	;	91	[	123	{
C	28	X	60	<	92	\	124	
D	29	/	61	=	93	]	125	}
E	30	\	62	>	94	^	126	~
F	31	+	63	?	95	-	127	(blank)

Hexa- decimal code	80 – 9F		A0 – BF		C0 – DF		E0 – FF	
	code	character	code	chracter	code	character	code	character
0	128	Ç	160	á	192	■	224	α
1	129	ü	161	í	193	■	225	β
2	130	é	162	ó	194	■	226	Γ
3	131	á	163	ú	195	■	227	π
4	132	ä	164	ñ	196	■	228	Σ
5	133	à	165	ñ	197	■	229	σ
6	134	â	166	ä	198	■	230	μ
7	135	ç	167	ö	199	■	231	γ
8	136	é	168	¿	200	■	232	Φ
9	137	ë	169	┌	201	■	233	θ
A	138	è	170	└	202	■	234	Ω
B	139	ï	171	½	203	▧	235	ó
C	140	ì	172	¼	204	▨	236	∞
D	141	í	173	ì	205	▩	237	φ
E	142	Ä	174	≪	206	▴	238	ε
F	143	Å	175	≫	207	▾	239	∩
0	144	É	176	À	208	▹	240	≡
1	145	æ	177	ā	209	⌘	241	±
2	146	Æ	178	ī	210	⌘	242	≥
3	147	ó	179	î	211	■	243	≤
4	148	ö	180	ō	212	■	244	∩
5	149	ò	181	õ	213	■	245	∩
6	150	ú	182	Ů	214	■	246	÷
7	151	ù	183	ü	215	▣	247	≈
8	152	ÿ	184	π	216	Δ	248	°
9	153	ÿ	185	ij	217	‡	249	•
A	154	Ü	186	¾	218	ω	250	•
B	155	Ć	187	~	219	■	251	√
C	156	£	188	◊	220	■	252	n
D	157	¥	189	‰	221	■	253	²
E	158	₣	190	¶	222	■	254	■
F	159	f	191	§	223	■	255	■

### Characters whose character code consists of 2 bytes

Characters of codes 1 to 31 (decimal) in the above table have 2-byte character codes. Their codes in the table should be preceded by the code 1 and the codes listed in the table should be added by 64 (decimal).

#### Input/output of character codes

Input from the keyboard

Normal characters ..... 1-byte code is input.

Example: Code 65 (decimal) for the character "A"

2-byte code characters ..... 1 and the other code are input.

Example: Code 1 and 67 for the character "♥"

#### Output using CHR\$ function

Normal characters ..... 1-byte code is used as a parameter.

Example: CHR\$(66) for the character "B"

2-byte code characters ..... 2 CHR\$ functions are used, of which the

first one is CHR\$(1), and the following one is a CHR\$ function using the above listed code as a parameter.

Example: CHR\$(1); CHR\$(68) for the character "♦"

## APENDICE A

## Descripción resumida de Microprocesadores

### PANORAMA ACTUAL DEL MERCADO DE LOS MICROPROCESADORES

Hoy en día, existen bastantes fabricantes americanos, japoneses y europeos que construyen microprocesadores, ya sea con versiones propias o actuando como “segundas fuentes” de modelos que han obtenido éxito a nivel mundial. Pero el marchamo que caracteriza a toda la microelectrónica es “la constante innovación y mejora de productos” tanto en orden al *hardware*, con la ayuda de las modernas técnicas VLSI de integración a gran escala, como al *software*, en un intento de normalizar y simplificar la programación de sistemas microcomputadores.

Tras la fase inicial de los microprocesadores de 4 bits, siguió la de 8 y después con la tecnología NMOS, la de la 3ª generación, potentes, flexibles y veloces.

La expansión de la aplicación de la microinformática a temas cada vez más exigentes dió lugar a lo microprocesadores de 16 bits y, últimamente, a los de 32. En estos momentos, la parte más importante del mercado es absorbida por lo microprocesadores de 8 bits, que son el objeto de este libro; pero la filosofía de utilización es similar en todos los tipos y así, conociendo bien uno de ellos, se puede comprender y aplicar cualquier otro fácilmente.

En capítulos precedentes se ha intentado explicar la teoría y la práctica sobre el diseño de sistemas microcomputadores usando una conocida familia, la 6500 de MOS TECHNOLOGY, Rockwell y Synertek. Con el presente Apéndice, se trata de proporcionar al lector los elementos básicos de otros microprocesadores importantes que, disponiendo de fa-

milias microcomputadoras parecidas a la que se ha estudiado, puedan servir de ayuda en la selección y aplicación de sus componentes.

En la tabla “Selección de microprocesadores y características generales”, se presentan unos cuantos modelos de relieve mundial. Se han elegido, para su descripción en el presente apartado, el Z-80, el 8085 y el 6800, considerados como los más populares a nivel mundial. Se comienza con el Z-80, que ha sido el microprocesador más vendido en 1981 y que entraña una complejidad superior (aunque no es difícil) frente a los restantes.

Por último, debe tenerse en cuenta que, *para cada aplicación y en cada situación, existe un microprocesador mejor que los demás, aunque, a priori, ninguno es mejor que otro.* Por lo tanto, a la hora de seleccionar una familia que implemente un sistema microcomputador, no sólo hay que valorar las características técnicas, sino, también, todas las circunstancias que le acompañan (precio, distribución, información, plazos de entrega, asesoramiento, soportes de *hardware* y *software*, etc.).

### EL MICROPROCESADOR Z-80

#### INTRODUCCION

El Z-80 es uno de los microprocesadores más importantes y usados de entre los correspondientes a la tercera generación. Su nombre deriva de su constructor, Zilog (Z), y, seguramente, de su descendencia de los microprocesadores de Intel 8008 y 8080, del que se considera una versión muy mejorada. Como ya se ha dicho, en 1981 fue el microprocesador más vendido en todo el mundo.

Dispone de un juego de instrucciones muy amplio y compatible con el del 8080.

El Z-80 tiene capacidad para funcionar a una frecuencia de reloj de 4 MHz, aunque dispone de versiones más económicas, con este parámetro más reducido, que simplifican su empleo.

#### ARQUITECTURA DEL Z-80

De forma similar a la mayoría de los microprocesadores, dispone de una serie de registros internos, que caracterizan su funcionamiento y que se describen seguidamente.

SELECCION DE MICROPROCESADORES Y CARACTERISTICAS GENERALES

Modelo	Fabricante	Segundas fuentes	Tamaño palabra	Tecnología	Capacidad memoria	Número instruc.	Alimentación
6502	MOS Tech.	Rockwell y Synertek	8	NMOS	64 K	56	5 V
6800	Motorola	AMI, Fairchild Fujitsu, Thomson-CSF/Secosem	8	NMOS	64 K	72	5 V
8085	Intel	AMD, NEC, Oki, Siemens y Toshiba	8	NMOS	64 K	72	5 V
Z-80	Zilog	Mostek, NEC, SGS-ATES y Sharp	8	NMOS	64 K	158	5 V
2650	Signetics		8	NMOS	32 K	75	5 V
8060	National	Western Digital	8	NMOS	64 K	46	5 V
F8	Fairchild	Mostek y Motorola	8	NMOS	64 K	67	5 y 12 V
990	Texas		8	NMOS	64 K	58	
8086	Intel	Siemens, Fujitsu, NEC, AMD	16	HMOS	1024 K	250	5 V
6800	Motorola	EFCIS, Hitachi, Mostek, Philips, Rockwell y Signetics	16	NMOS	16 M	61	5 V
9900	Texas	AMI, ITT e Intermetall	16	NMOS	64 K	65	12 y 5 V
Z 8001/2	Zilog	AMD, SGS y Sharp	16				

Acumulador y Unidad Lógico-Aritmética

El Acumulador es un registro de 8 bits de especial relieve, puesto que su empleo es requerido por la ALU del microprocesador en la mayoría de las operaciones lógicas y aritméticas que lleva a cabo.

En general, el Acumulador proporciona un operando a la ALU, cargándose con el resultado de la operación una vez realizada, como se expresa gráficamente en la Fig. A-1.

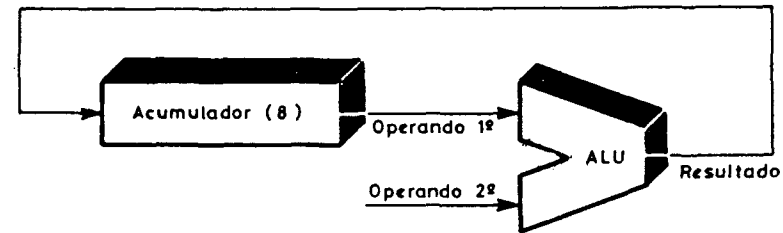


Fig. A-1.- El Acumulador proporciona un operando a la ALU y se carga con el resultado de la operación.

Esta forma particular de trabajo de los microprocesadores permite expresar las instrucciones usando una sola dirección, que indica la situación del 2º operando. El 1º operando y el resultado se localizan, siempre, en el Acumulador.

Registros generales de trabajo

El Z-80 dispone de 6 registros de 8 bits que reciben el nombre de B, C, D, E, H y L. Normalmente, están agrupados por parejas, como se muestra en la Fig. A-2.

Registros de 8 bits

B	C
D	E
H	L

Fig. A-2.- Otros registros de trabajo, además del Acumulador.

Los registros de trabajo se emplean, en lugar de posiciones de memoria, para almacenar datos y recuperarlos de una forma más directa y rápida.

Existen instrucciones del Z-80 que permiten trabajar con "parejas" de registros. Es decir, se puede utilizar conjuntamente el registro B con el C, el registro D con el E y finalmente el H con el L. Esto supone manejar informaciones de 16 bits, lo que aumenta sustancialmente la precisión de las operaciones aritméticas, así como el direccionamiento de todo el mapa de memoria, puesto que el bus de direcciones del Z-80, como se explicará más adelante, consta de 16 líneas.

El conjunto total de registros principales de trabajo del Z-80 está formado por los 8 que aparecen en la Fig. A-3.

A	F
B	C
D	E
H	L

Fig. A-3.— Conjunto completo de los registros principales de trabajo del Z-80.

El registro F, asociado al Acumulador, se denomina registro de "flags", siendo éstos los bits que señalizan la producción de ciertas condiciones especiales.

El bit de más peso del registro F indica el signo del resultado de la operación aritmética que se acaba de efectuar. Si vale 1 significa que es negativo, y si es 0, positivo. El flag Z (cero) vale 1 si el resultado de la operación anterior ha sido de todos ceros. H es el *flag* que pasa a 1 cuando hay acarreo en el 4º bit. P/V es el *flag* que se emplea para expresar la paridad o el sobrepasamiento. C indica el acarreo en el bit de más peso, y N un número negativo. Véase la Fig. A-4.

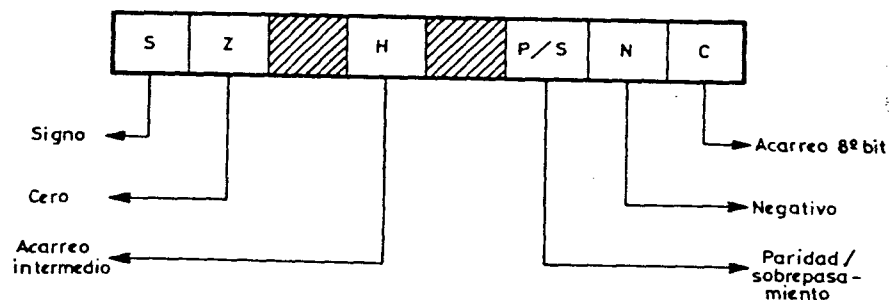


Fig. A-4.— Significado de los flags del registro F del Z-80.

## Los registros duplicados

El Z-80 tiene duplicados los registros A, F, B, C, D, E, H y L, recibiendo los nuevos la denominación A', F', B', C', D', E', H' y L'. Sin embargo, el microprocesador, en cada momento, sólo puede usar 8 del total de los 16 registros existentes, los principales, o los primados. La selección del tipo de registros se efectúa a través de dos instrucciones destinadas a este fin.

Existen dos causas que justifican la duplicación de los registros internos:

- 1ª) A veces, se precisa esta capacidad supletoria de almacenamiento, que puede ser conmutada rapidísimamente.
- 2ª) Para permitir una mayor flexibilidad en la manipulación de las interrupciones.

## Registros Índices

El Z-80 dispone de dos registros índices, designados IX e IY, que permiten disponer, en el repertorio de instrucciones del Z-80, del modo de direccionado "indexado", tan interesante para determinadas aplicaciones.

## Registro I (Interrupts)

Se emplea para implementar hasta 128 interrupciones diferentes. Proporciona el primer byte de la dirección de la rutina de servicio de interrupción, cuando se produce una de éstas.

## Registro R (Refresh)

Su misión es controlar la generación del refresco que necesitan las memorias dinámicas.

## Registro PC (Program Counter)

Se trata del Contador de Programa que, con 16 bits, señala la instrucción que va a ser ejecutada.

Si, por ejemplo, en la posición A251 se encuentra la instrucción LD A, (3FA0), que significa cargar el Acumulador con el contenido de la posición 3FA0, el PC actúa de la siguiente manera, durante la ejecución de esta instrucción:

1. El PC contiene la dirección A251.



2. El primer byte (código OP) de la instrucción LD A, (3FA0) se carga en la CPU.
3. El PC se incrementa a A252.
4. La decodificación del código de operación de la instrucción proporciona a la CPU el conocimiento de la misma, así como la necesidad de localizar, en las dos posiciones de memoria siguientes, la dirección de la memoria de datos donde se encuentra el operando.
5. El 2º byte de la instrucción, que direccionaba el PC y que constituye la mitad de la dirección de memoria en donde se encuentra el operando, se carga en la CPU.
6. El PC se incrementa a A253.
7. Se lee, en la CPU, el 3º byte de la instrucción.
8. El PC se incrementa a A254.
9. Se ejecuta la instrucción LD A, (3FA0).
10. Comienza una nueva instrucción desde la posición A254.

El PC puede cargarse con cualquier dirección (y no seguir su ritmo normal de incremento) empleando las instrucciones adecuadas al efecto.

**Registro SP (Stack Pointer)**

Se trata de un registro de 16 bits que direcciona cualquier parte de la memoria y se emplea para salvar el contenido de ciertos registros de la CPU en los casos en que se produzcan interrupciones o saltos a subrutinas que exigen el regreso al estado inicial del programa principal.

En la Fig. A-5 se ofrece el conjunto de los registros disponibles en el Z-80.

Registros generales		Registros especiales	
A (8)	F (8)	I (8)	R (8)
B (8)	C (8)	IX (16)	
D (8)	E (8)	IY (16)	
H (8)	L (8)	SP (16)	
A' (8)	F' (8)	PC (16)	
B' (8)	C' (8)		
D' (8)	E' (8)		
H' (8)	L' (8)		

Fig. A-5.- Conjunto general de los registros del Z-80.

**DIAGRAMA DE CONEXIONADO DEL Z-80**

Estudiada la configuración interna de la sección de registros, que es la que más interés tiene para el usuario puesto que a su través y mediante las correspondientes instrucciones se controlará la actuación del sistema, se describe, seguidamente, la distribución de las 40 patillas del Z-80, mostradas en la Fig. A-6.

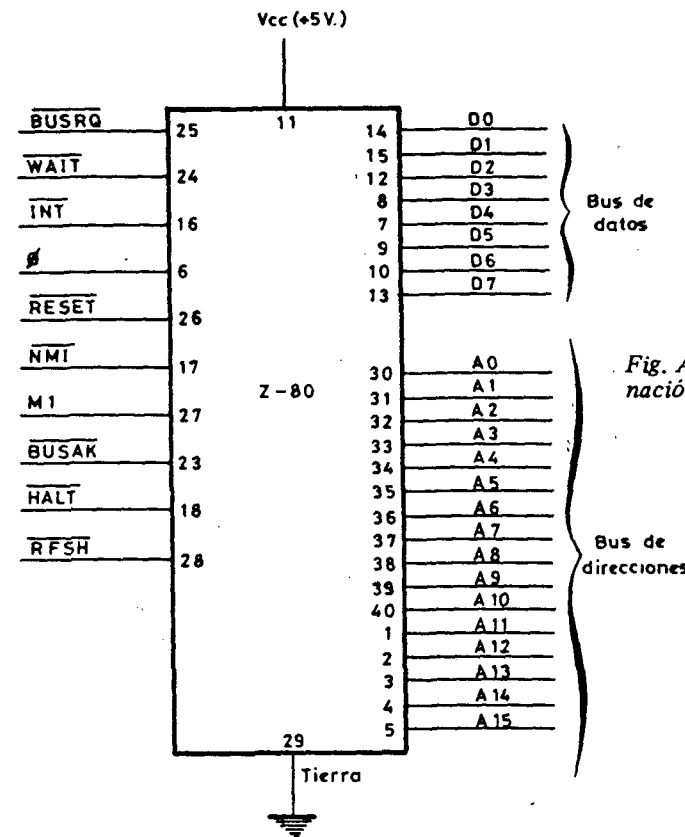


Fig. A-6.- Distribución y asignación de las patillas del Z-80.

La explicación del cometido de las patillas se ofrece a continuación, agrupando a las que tienen funciones similares.

**Bus de datos**

Está constituido por 8 patillas que se conectan a las 8 líneas bidireccionales que sacan e introducen datos de la CPU respecto a los demás módulos del sistema.

**Bus de direcciones**

Lo forman 16 líneas unidireccionales, por las que la CPU transmite la dirección con la que se va a llevar a cabo una operación.

**Alimentación**

Por la patilla 11 se aplica la tensión de 5 V; por la 29 se conecta a tierra.

**Señal de lectura ( $\overline{RD}$ )**

Cuando está a nivel lógico bajo, indica la realización de un ciclo de lectura de la memoria.

**Petición de memoria ( $\overline{MREQ}$ )**

Pasa a nivel bajo cuando la CPU está realizando un acceso a memoria (lectura o escritura).

**Señal de escritura ( $\overline{WR}$ )**

Se activa con nivel bajo cuando se efectúa una operación de escritura de memoria.

**Petición de I/O ( $\overline{IORQ}$ )**

Se activa con nivel bajo cuando se realiza una operación de I/O.

La lectura y escritura de la memoria y las I/O lleva aparejado el tratamiento de las señales  $\overline{RD}$ ,  $\overline{WR}$ ,  $\overline{MREQ}$  y  $\overline{IORQ}$ .

Conjuntamente con  $\overline{MI}$ , esta señal se emplea, también, para reconocimiento de interrupciones.

**Control de DMA ( $\overline{BUSRQ}$  y  $\overline{BUSAK}$ )**

Estas dos patillas son manipuladas por los elementos encargados de proveer de "Acceso Directo a Memoria" (DMA) al sistema. La DMA consiste en la transferencia directa de datos entre la memoria y los elementos de I/O.

$\overline{BUSRQ}$  es la señal de petición de bus y  $\overline{BUSAK}$  indica que la CPU ha recibido la petición de control del sistema de buses.

**Señal de refresco para memorias dinámicas ( $\overline{RFSH}$ )**

Su activación indica la disponibilidad de la dirección del registro de refresco, R, en el bus de direcciones.

**Señal de paro ( $\overline{HALT}$ )**

Su activación indica que la CPU no ejecuta ninguna instrucción.

**Señal de espera ( $\overline{WAIT}$ )**

Se trata de una patilla de entrada, cuya activación detiene al Z-80, permitiendo acoplar la velocidad de elementos de memoria y de I/O más lentos que la CPU.

**Señal de estado ( $\overline{MI}$ )**

Cuando se activa, indica que se está realizando un ciclo de búsqueda de una instrucción.

**Señal de Reset ( $\overline{RESET}$ )**

Se trata de una señal de entrada que se usa en la conexión de la alimentación o cuando se desea regresar a un estado conocido. La activación de  $\overline{RESET}$  origina la puesta a cero del PC y la ejecución de instrucciones desde la posición cero de la memoria.

**Interrupción no mascarable ( $\overline{NMI}$ )**

Su activación provoca la ejecución de una interrupción no enmascarable, es decir, de atención prioritaria y segura. Al producirse una interrupción de este tipo, la CPU se detiene, salvándose el estado del PC en la memoria controlada por el *Stack Pointer* y cargándose el PC con una dirección de memoria donde comienza la rutina de atención a dicha interrupción. Al final de la rutina de interrupción, y tras una instrucción RTI (Retorno de Interrupción), se devuelve, desde el *Stack*, el estado inicial de la CPU, regresando al programa principal.

**Señal de interrupción ( $\overline{INT}$ )**

Es una entrada para producir una interrupción mascarable.

### Señal de reloj ( $\phi$ )

Se trata de una entrada para la señal de reloj que establezca la frecuencia de trabajo de la CPU. El Z-80 puede trabajar hasta con una onda cuadrada de 4 MHz, siendo lo más frecuente trabajar a 1 MHz en los casos en que no se requiere mucha velocidad.

### LAS INSTRUCCIONES DEL Z-80

El juego de instrucciones del Z-80 es muy extenso y, al igual que en la mayoría de los microprocesadores, las instrucciones pueden clasificarse en varios grupos típicos:

- De carga.
- Lógico-Aritméticas.
- Saltos, llamadas a subrutinas y retornos.
- Rotaciones y desplazamientos.
- Instrucciones de puesta a 1, puesta a 0 y testado de bits.
- Instrucciones de Entrada/Salida.
- Otros tipos diversos.

A continuación, se ofrece una descripción general de las instrucciones que cada grupo comprende.

#### Instrucciones de carga

En realidad, en este grupo se encuentran todas las instrucciones de transferencia de información (1 ó 2 bytes) entre la memoria y los registros de la CPU, o de éstos entre sí. A todo este conjunto de instrucciones se les asigna el nemónico "LD" (Load: carga).

Una instrucción típica de este grupo puede ser la siguiente: LD A, B, con la que se transfiere el contenido del registro B (fuente) al A (destino). Téngase en cuenta que el registro fuente, del que procede la información, no cambia su contenido una vez efectuada la instrucción.

Otra instrucción típica de este grupo es: LD A, (017F), con la que el contenido de la posición de memoria 017F (por eso se coloca entre paréntesis) se transfiere al Acumulador. Tras la ejecución de la instrucción, el contenido de la posición de memoria no queda afectado.

La instrucción contraria a la anterior sería LD (017F), A, con la que el contenido del Acumulador se transfiere a la posición 017F.

Una instrucción de carga inmediata puede ser: LD A, 2F, con la que el Acumulador se carga con el dato 2F.

También pueden transferirse 16 bits, en una sola instrucción, usando los registros por parejas (AF, BC, DE y HL), así como los que, intrínsecamente, disponen de 16 bits, como el SP, IX e IY. Así, por ejemplo, la instrucción LD HL, 7F05 carga el dato de 16 bits 7F05 en la pareja de registros HL. Debe tenerse presente, en estas instrucciones, que el primer byte del operando es el de menos peso en la estructura del código máquina de la instrucción y, en este caso, será el que se cargue en el registro L. De esta forma, la instrucción del ejemplo contará con tres bytes, el primero de los cuales es el código de operación y los otros dos el operando, tal como se expresa en la Fig. A-7.

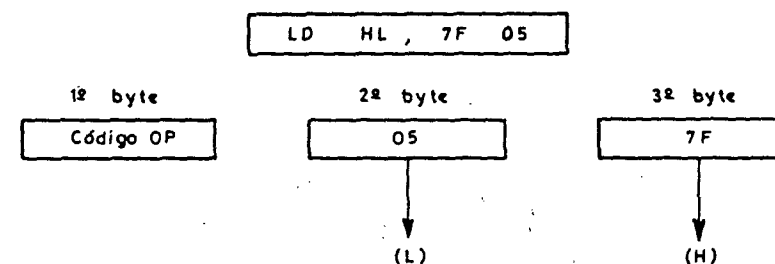


Fig. A-7.- Composición y trasvase de los dos bytes del operando a los registros de la CPU, en la instrucción LD HL, 7F05.

Otra instrucción, que envuelve una transferencia de 16 bits, puede ser LD (A50E), BC, que transfiere el contenido de las posiciones A50F y A50E, respectivamente, a la pareja de registros BC.

#### Instrucciones aritméticas y lógicas

En el caso de las instrucciones aritméticas que manipulan datos de 8 bits, el Acumulador es, siempre, el registro destino. Se ofrecen algunos ejemplos de este tipo de instrucciones.

ADD A, H: Se suman los contenidos de los registros A y H y el resultado se deposita en A.

ADD A, (HL): Se suma al valor de A, el contenido de la posición de memoria direccionada por la pareja de registros HL. El resultado se deposita en A.

SUB 2A: Se resta el valor 2A del Acumulador y el resultado se deja en A.

Las sumas y restas de datos de 16 bits exigen que la pareja de registros HL, o los registros IX o IY, se empleen como registro destino. Ejemplos:

**SBC HL, DE:** Al contenido de la pareja de registros HL, se resta el correspondiente a la pareja DE y, también, se resta el valor del carry. El resultado se deposita en HL.

Cuando se usan números con signo, el bit de más peso indica el signo, siendo un 0 el que corresponde a los números positivos y un 1 a los negativos, quienes vienen expresados en forma de complemento a dos.

En las operaciones aritméticas, ha de tenerse un especial cuidado al analizar los resultados, debiéndose tener siempre muy en cuenta el estado de los *flags* del registro F, que señalizan situaciones que hay que corregir con *software*.

En cuanto a las operaciones lógicas, sólo pueden realizarse con 8 bits; el registro A siempre actúa como destino. Las operaciones lógicas más representativas son la AND, OR y la XOR (OR exclusiva).

### Instrucciones de saltos, llamadas y retornos

Las instrucciones de salto condicional obligan al PC a cambiar su contenido por el que se indica detrás del código OP de la instrucción.

Las instrucciones de "llamada a subrutina" (CALL), también alteran el valor del PC, pero, antes de producirse el salto, el estado de la CPU se salva (en la zona de memoria controlada por el Stack Pointer) para cuando se desee regresar al programa principal, mediante el uso de la instrucción de "retorno" (RET).

Finalmente, también existen instrucciones de salto o llamadas condicionales, que no se llevan a cabo si no se cumple una condición impuesta, relacionada con el estado de alguno de los *flags*.

Además de salvar automáticamente al PC en las instrucciones de llamada y de salto con retorno, el *Stack Pointer* puede emplearse, también, para guardar, provisionalmente, el contenido de los registros de la CPU, mediante las instrucciones PUSH (empujar) y POP (sacar). El *Stack Pointer* no es más que un registro-contador que apunta una zona de memoria en la que se guardan y salvan registros de la CPU.

### Instrucciones de rotación y desplazamiento

Las instrucciones de rotación afectan a los registros de la CPU o a posiciones de memoria. Consisten en la rotación de los contenidos de los bits a la derecha o a la izquierda. La rotación puede afectar sólo a los 8

bits del registro o posición de memoria o a 9, cuando interviene el carry. Véase la Fig. A-8.

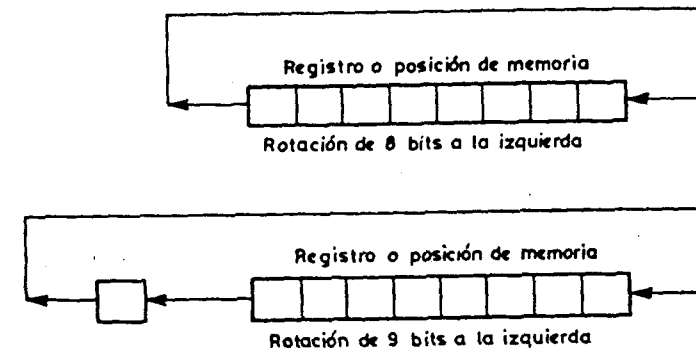


Fig. A-8.- Los dos tipos posibles de instrucciones de rotación.

En las instrucciones de desplazamiento, el último bit se pierde. En los desplazamientos lógicos, el primer bit que se desplaza es repuesto con un cero y el último se pierde. Si el desplazamiento es aritmético, el bit de signo permanece invariable y el último se pierde. Véase la Fig. A-9.

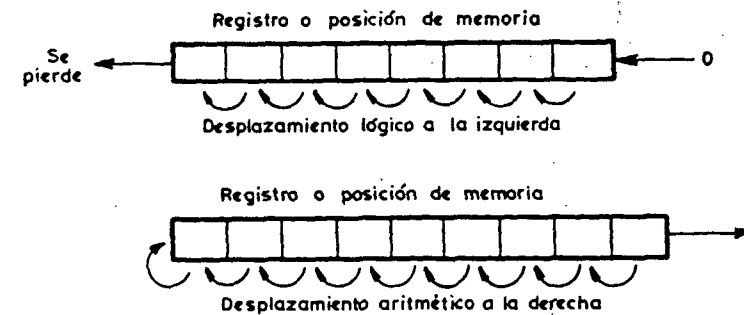


Fig. A-9.- Comportamiento de las instrucciones de desplazamiento.

### Instrucciones de puesta a 1, puesta a 0 y testado de bits

Permiten poner a 1, a 0 o conocer el estado de cualquier bit de los registros de la CPU o de una posición de memoria.

Por ejemplo, con la instrucción "SET 3, C" el bit 3 del registro C se coloca en 1. Con la instrucción "RESET 7, B" el bit de más peso del registro B se pone en 0.

Para conocer el estado de un bit, se usa la instrucción de nemónico BIT, que pasa al carry el bit deseado del registro o de la posición de memoria seleccionada. Así "BIT 6, A" coloca el bit 6 del Acumulador en el carry.

### Instrucciones de Entrada/Salida y diversas

Para sacar o introducir información de la CPU o desde el exterior, se emplean instrucciones de lectura y escritura de las posiciones de memoria asignadas a los dispositivos de Entrada/Salida.

Así, la instrucción "RD A, (01A2)", lee la información del dispositivo ubicado en la dirección 01A2 y la traslada al Acumulador. La instrucción opuesta será WR (01A2), A.

Finalmente, existen otras instrucciones particulares que se emplean ocasionalmente en los programas y que se recomienda estudiar directamente sobre el juego de instrucciones del Z-80.

### MODOS DE DIRECCIONAMIENTO DE LAS INSTRUCCIONES DEL Z-80

Los modos de direccionamiento consisten en las diferentes formas que tienen las instrucciones para indicar la situación de sus operandos.

En el Z-80, además de los direccionamientos de su predecesor el 8080, existen otros nuevos que se describen a continuación:

1. Direccionamiento implicado.
2. Direccionamiento inmediato.
3. Direccionamiento por registro.
4. Direccionamiento indirecto por registro.
5. Direccionamiento extendido.
6. Direccionamiento por página cero.
7. Direccionamiento relativo.
8. Direccionamiento indexado.
9. Direccionamiento de bit.

### Direccionamiento implicado

Cuando el operando está implicado directamente en la instrucción. Por ejemplo, "SCF", que pone el *flag* de acarreo (carry) en 1 (set).

### Direccionamiento inmediato

Cuando el operando se especifica directamente detrás del código de operación de la instrucción. Ejemplos de esta forma de direccionado son las instrucciones "LD A, FF", con la que se carga el dato FF en el Acumulador, o bien, "LD HL, FFFF", con la que se cargan los registros H y L con todos sus bits en 1.

### Direccionamiento por registro

Cuando el operando queda definido por el contenido de un registro. Así por ejemplo, "ADD A, B", que suma el contenido de los registros A y B, dejando depositado el resultado en A.

### Direccionamiento indirecto por registro

Este direccionado se usa cuando el operando está contenido en una posición de memoria cuya dirección se especifica por el contenido de una pareja de registros. Ejemplo: "ADD A, (HL)", que suma el valor de A con el contenido de la dirección especificada por el valor guardado por H y L.

### Direccionamiento extendido

Cuando se proporciona directamente la posición de memoria en la que se encuentra el operando. Ejemplo: "ADD A, (5050)", con la que el contenido de la posición 5050 se suma a A. Otro ejemplo para datos de 16 bits puede ser la instrucción LD HL, (5050), con la que se carga el contenido de la posición 5050 con el del registro L, y el de la posición 5051 con el de H.

### Direccionamiento por página cero

Sólo se emplea para la instrucción RST (RESTART). Esta instrucción es similar a las de llamada a subrutina, del tipo CALL. Según el código de RST, el contador de programa salta a las posiciones de la página cero: 0, 8, 16, 24, 32, 40, 48 y 56.

Mientras que las instrucciones CALL constaban de 3 bytes, las RST sólo se definen con 1 byte.

### Direccionamiento relativo

Se usa para los saltos en los cuales, al valor del PC se suma el del 2º byte (offset) de la instrucción, que sigue al código OP. Así por ejemplo, la instrucción JR 65 obliga a saltar 65<sub>16</sub> posiciones de memoria al PC.

### Direccionamiento indexado

El operando se halla en una dirección de memoria que se obtiene sumando el byte que se incluye en la instrucción como offset a uno de 19 registros Índice (IX o IY). Ejemplo: ADD A, (IY + 31). El sumando que hay que añadir al Acumulador está en la dirección especificada por el contenido de IY más 31<sub>16</sub>, que actúa como offset. El código OP de estas instrucciones es de dos bytes, por lo que la instrucción consta de 3 bytes.

### Direccionamiento por bit

Se usa en las instrucciones de puesta a 1, a 0 y de testado de un bit determinado de un registro o posición de memoria.

## JUEGO DE INSTRUCCIONES DEL Z-80

A continuación, se ofrece una exposición de las instrucciones del Z-80 en forma simplificada y en la que se han utilizado los siguientes símbolos:

- r: registro (0 = B, 1 = C, 2 = D, 3 = E, 4 = H, 5 = L, 7 = A).
- r': Igual que para r, pero primados.
- ss: Pareja de registros (0 = BC, 1 = DE, 2 = HL y 3 = SP).
- b: Campo de los bits desde 0 a 7.
- c: Condición de los flags (0 = NZ, 1 = Z, 2 = NC, 3 = NC, 3 = C, 4 = PO, 5 = PE, 6 = P y 7 = M).
- d: Desplazamiento indexado de 127 a -128.
- e: Salto de desplazamiento relativo de 127 a -128.
- n: Valor inmediato o dirección.
- t: Campo para la instrucción RST.
- dd: Pareja de registros (0 = BC, 1 = DE, 2 = HL, 3 = SP).
- pp: Pareja de registros (0 = BC, 1 = DE, 2 = HL, 3 = SP).
- gg: Pareja de registros (0 = BC, 1 = DE, 2 = IY, 3 = SP).

NEMONICO	EXPRESION DESCRIPTIVA	FLAGS AFECTADOS
ADC HL, ss	HL + ss + CY → HL	S, Z, P/V y C
ADC A, r	A + r + CY → A	S, Z, P/V y C
ADC A, n	A + n + CY → A	S, Z, P/V y C
ADC A, (HL)	A + (HL) + CY → A	S, Z, P/V y C
ADC A, (IX + d)	A + (IX + d) + CY → A	S, Z, P/V y C
ADC A, (IY + d)	A + (IY + d) + CY → A	S, Z, P/V y C
ADD A, n	A + n → A	S, Z, P/V y C
ADD A, r	A + r → A	S, Z, P/V y C
ADD A, (HL)	A + (HL) → A	S, Z, P/V y C
ADD A, (IX + d)	A + (IX + d) → A	S, Z, P/V y C
ADD A, (IY + d)	A + (IY + d) → A	S, Z, P/V y C
ADD HL, ss	HL + ss → HL	C
ADD IX, pp	IX + pp → IX	C
ADD IY, rr	IY + rr → IY	C
AND r	A AND r → A	S, Z, P/V y C = 0
AND n	A AND n → A	S, Z, P/V y C = 0
AND (HL)	A AND (HL) → A	S, Z, P/V y C = 0
AND (IX + d)	A AND (IX + d) → A	S, Z, P/V y C = 0
AND (IY + d)	A AND (IY + d) → A	S, Z, P/V y C = 0
BIT b, r	Testado del bit b de r	S, Z y P/V
BIT b, (HL)	Testado bit b de (HL)	S, Z y P/V
BIT b, (IX + d)	Testado bit b de (IX + d)	S, Z y P/V
BIT b, (IY + d)	Testado bit b de (IY + d)	S, Z y P/V
CALL cc, nn	Salto subrutina en nn, si cumple cc	
CALL nn	Llamada incondicional a nn	
CCF	Complementación del carry	C
CP r	Comparación A con r	S, Z, P/V y C
CP n	Comparación A con n	S, Z, P/V y C
CP (HL)	Comparación A con (HL)	S, Z, P/V y C
CP (IX + d)	Comparación A con (IX + d)	S, Z, P/V y C
CP (IY + d)	Comparación A con (IY + d)	S, Z, P/V y C
CPD	Comparar bloque sin repetir	S, Z, P/V
CPDR	Comparar bloque repitiendo	S, Z, P/V
CPI	Comparar bloque sin repetir	S, Z, P/V
CPIR	Comparar bloque repitiendo	S, Z, P/V
CPL	Complemento a 1 de A	
DAA	Ajuste decimal de A	S, Z y P/V
DEC r	Decrementar r	S, Z y P/V

NEMONICO	EXPRESION DESCRIPTIVA	FLAGS AFECTADOS.
DEC (HL)	Decrementar (HL)	S, Z y P/V
DEC (IX + d)	Decrementar (IX + d)	S, Z y P/V
DEC (IY + d)	Decrementar (IY + d)	S, Z y P/V
DEC IX	Decrementa IX	
DEC IY	Decrementa IY	
DEC ss	Decrementa la pareja de registros	
DI	Prohíbe interrupciones	
DJNZ e	Decrementa B y JR si B ≠ 0	
EI	Permite interrupciones	
EX (SP), HL	Intercambia (SP) y HL	
EX (SP), IX	Intercambia (SP) e IX	
EX (SP), IY	Intercambia (SP) e IY	
EX AF, A'F'	Pone activos a AF primados	
EX DE, HL	Intercambia DE y HL	
EXX	Pone activo a B-L primado	
HALT	Paro	
IM 0	Pone interrupciones en modo 0	
IM 1	Pone interrupciones en modo 1	
IM 2	Pone interrupciones en modo 2	
IN A, (n)	Carga A con la entrada n	
IN r, (C)	Carga r con entrada desde (C)	S, Z y P/V
INC r	Incrementa r	S, Z y P/V
INC (HL)	Incrementa (HL)	S, Z y P/V
INC (IX + d)	Incrementa (IX + d)	S, Z y P/V
INC (IY + d)	Incrementa (IY + d)	S, Z y P/V
INC IX	Incrementa IX	
INC IY	Incrementa IY	
INC ss	Incrementa pareja de registros ss	
IND	Bloque I/O de entrada desde (C) (HL) ← (C); B ← B-1; HL ← HL-1	S, Z y P/V
INDR	Bloque entrada I/O, repite hasta B = 0 (HL) ← (C); B ← B-1; HL ← HL-1	S, Z y P/V
INI	Bloque entrada I/O desde (C) (HL) ← (C); B ← B-1; HL ← HL-1	S, Z y P/V
INIR	Bloque entrada I/O, repite hasta B = 0 (HL) ← (C); B ← B-1; HL ← HL-1	S, Z y P/V
JP (HL)	Salto incondicional a (HL)	
JP (IX)	Salto incondicional a (IX)	
JP (IY)	Salto incondicional a (IY)	

NEMONICO	EXPRESION DESCRIPTIVA	FLAGS AFECTADOS
JP cc, nn	Salto a nn si cc	
JP nn	Salto incondicional a nn	
JR C, e	Salto relativo si hay carry	
JR e	Salto relativo incondicional	
JR NC, e	Salto relativo si no hay carry	
JR NZ, e	Salto relativo si Z = 0	
JR Z, e	Salto relativo si Z = 1	
LD A, (BC)	Carga A con (BC)	
LD A, (DE)	Carga A con (DE)	
LD A, I	Carga A con I	S, Z y P/V
LD A, (nn)	Carga A con el contenido de nn	
LD A, R	Carga A con R	S, Z y P/V
LD (BC), A	Almacena A en (BC)	
LD (DE), A	Almacena A en (DE)	
LD (HL), n	Almacena n en (HL)	
LD dd, nn	Carga un par de registros con nn	
LD dd, (nn)	Carga un par de registros con posición nn	
LD HL, (nn)	Carga (HL) con posición nn	
LD (HL), r	Almacena r en (HL)	
LD I, A	Carga I con A	
LD IX, (nn)	Carga IX con nn	
LD IX, nn	Carga IX con posición nn	
LD (IX + d), r	Almacena r en (IX + d)	
LD (IX + d), n	Almacena n en (IX + d)	
LD IY, nn	Carga IY con nn	
LD IY, (nn)	Carga IY con posición nn	
LD (IY + d), n	Almacena n en (IY + d)	
LD (IY + d), r	Almacena r en (IY + d)	
LD (nn), A	Almacena A en la posición nn	
LD (nn), dd	Almacena par de reg. en posición nn	
LD (nn), HL	Almacena HL en posición nn	
LD (nn), IX	Almacena IX en posición nn	
LD (nn), IY	Almacena IY en posición nn	
LD r, A	Carga R con A	
LD r, r'	Carga r con r'	
LD r, n	Carga r con n	
LD r, (HL)	Carga r con (HL)	
LD r, (IX + d)	Carga r con (IX + d)	

NEMONICO	EXPRESION DESCRIPTIVA	FLAGS AFECTADOS
LD r, (IY + d)	Carga r con (IY + d)	
LD SP, HL	Carga SP con HL	
LD SP, IX	Carga SP con IX	
LD SP, IY	Carga SP con IY	
LDD	Carga bloque, sin repetir	P/V
LDDR	Carga bloque, repite	P/V
LDI	Carga bloque, sin repetir	P/V
LDIR	Carga bloque, repite	P/V
NEG	Niega A (lo pone en complemento a 2)	S, Z, P/V y C
NOP	No opera	
OR r	A OR r → A	S, Z, P/V y C
OR n	A OR n → A	S, Z, P/V y C
OR (HL)	A OR (HL) → A	S, Z, P/V y C
OR (IX + d)	A OR (IX + d) → A	S, Z, P/V y C
OR (IY + d)	A OR (IY + d) → A	S, Z, P/V y C
OTDR	Bloque salida, repite hasta B = 0 (C) ← (HL); B ← B-1; HL ← HL-1	S, Z y P/V
OTIR	Bloque salida, repite hasta B = 0 (C) ← (HL); B ← B-1; HL ← HL + 1	S, Z y P/V
OUT (C), r	Salida r a (C)	
OUT (n), A	Salida A a la puerta n	
OUTD	Salida bloque, no repite (C) ← (HL); B ← B-1; HL ← HL-1	S, Z y P/V
OUTI	Salida bloque, no repite (C) ← (HL); B ← B-1; HL ← HL + 1	S, Z y P/V
POP IX	Saca IX desde stack	
POP IY	Saca IY desde stack	
POP qq	Saca qq desde stack	
PUSH IX	Mete IX al stack	
PUSH IY	Mete IY al stack	
PUSH qq	Mete qq al stack	
RES b, r	Reset del bit b de r	
RES b, (HL)	Reset del bit b de (HL)	
RES b, (IX + d)	Reset del bit b de (IX + d)	
RES b, (IY + d)	Reset del bit b de (IY + d)	
RET	Retorno desde subrutina	
RET cc	Retorno desde subrutina si cc	
RETI	Retorno desde interrupción	

NEMONICO	EXPRESION DESCRIPTIVA	FLAGS AFECTADOS
RETN	Retorno desde interrupción no mascarable	
RL r	Rotación izq. sin carry de r	S, Z, P/V y C
RL (HL)	Rotación izq. de (HL) sin carry	S, Z, P/V y C
RL (IX + d)	Rotación izq. (IX + d) sin carry	S, Z, P/V y C
RL (IY + d)	Rotación izq. (IY + d) sin carry	S, Z, P/V y C
RLA	Rotación izq. de A sin carry	C
RLC r	Rotación izq. de r con carry	S, Z, P/V y C
RLC (HL)	Rotación izq. de (HL) con carry	S, Z, P/V y C
RLC (IX + d)	Rotación izq. de (IX + d) con carry	S, Z, P/V y C
RLC (IY + d)	Rotación izq. de (IY + d) con carry	S, Z, P/V y C
RLCA	Rotación izq. de A con carry	C
RLD	Rotación izq. dígito bcd (HL)	S, Z, P/V
RR r	Rotación der. de r sin carry	S, Z, P/V y C
RR (HL)	Rotación der. de (HL) sin C	S, Z, P/V y C
RR (IX + d)	Rotación der. de (IX + d) sin C	S, Z, P/V y C
RR (IY + d)	Rotación der. de (IY + d) sin C	S, Z, P/V y C
RRA	Rotación der. de A sin C	C
RRC r	Rotación der. de r con C	S, Z, P/V y C
RRC (HL)	Rotación der. de (HL) con C	S, Z, P/V y C
RRC (IX + d)	Rotación der. de (IX + d) con C	S, Z, P/V y C
RRC (IY + d)	Rotación der. de (IY + d) con C	S, Z, P/V y C
RRCA	Rotación der. de A con C	C
RRD	Rotación der. dígito bcd (HL)	S, Z y P/V
RST p	Comenzar en posición p	
SBC A, r	A - r - C → A	S, Z, P/V y C
SBC A, n	A - n - C → A	S, Z, P/V y C
SBC A, (HL)	A - (HL) - C → A	S, Z, P/V y C
SBC A, (IX + d)	A - (IX + d) - C → A	S, Z, P/V y C
SBC A, (IY + d)	A - (IY + d) - C → A	S, Z, P/V y C
SBC HL, ss	HL - ss - C → HL	S, Z, P/V y C
SCF	Pone a 1 C	C
SET b, (HL)	Pone a 1 bit b de (HL)	
STE b, (IX + d)	Pone a 1 bit b de (IX + d)	
SET b (IY + d)	Pone a 1 bit b de (IY + d)	
SET b, r	Pone a 1 bit b de r	
SLA r	Desplaz. aritmét. izq. de r.	S, Z, P/V y C
SLA (HL)	Desplaz. aritmét. izq. de (HL)	S, Z, P/V y C



NEMONICO	EXPRESION DESCRIPTIVA	FLAGS AFECTADOS
SLA (IX + d)	Desplaz. aritmét. izq. de (IX + d)	S, Z, P/V y C
SLA (IY + d)	Desplaz. aritmét. izq. de (IY + d)	S, Z, P/V y C
SRA r	Desplaz. aritmét. der. de r	S, Z, P/V y C
SRA (HL)	Desplaz. aritmét. der. de (HL)	S, Z, P/V y C
SRA (IX + d)	Desplaz. aritmét. der. de (IX + d)	S, Z, P/V y C
SRA (IY + d)	Desplaz. aritmét. der. de (IY + d)	S, Z, P/V y C
SRL r	Deplaz. lógico der. de r	S, Z, P/V y C
SRL (HL)	Deplaz. lógico der. de (HL)	S, Z, P/V y C
SRL (IX + d)	Deplaz. lógico der. de (IX + d)	S, Z, P/V y C
SRL (IY + d)	Deplaz. lógico der. de (IY + d)	S, Z, P/V y C
SUB r	A - r → A	S, Z, P/V y C
SUB n	A - n → A	S, Z, P/V y C
SUB (HL)	A - (HL) → A	S, Z, P/V y C
SUB (IX + d)	A - (IX + d) → A	S, Z, P/V y C
SUB (IY + d)	A - (IY + d) → A	S, Z, P/V y C
XOR r	A EOR r → A	S, Z, P/V y C
XOR n	A EOR n → A	S, Z, P/V y C
XOR (HL)	A EOR (HL) → A	S, Z, P/V y C
XOR (IX + d)	A EOR (IX + d) → A	S, Z, P/V y C
XOR (IY + d)	A EOR (IY + d) → A	S, Z, P/V y C

**EL MICROPROCESADOR 8085**

**CARACTERISTICAS GENERALES DEL 8085**

El microprocesador 8085 es una versión mejorada del 8080 del mismo fabricante, el cual procede a su vez del 8008, que INTEL introdujo en 1973 para actuar como controlador de TRC para un equipo de DATA-POINT.

De acuerdo con la arquitectura general de todo microprocesador, el 8085 dispone, como puede apreciarse en la Fig. A-10, del contador de programa; buffers para la entrada y salida de información por los buses de direcciones y de datos; la ALU con el acumulador; el decodificador de instrucciones; el circuito de control y tiempos y el Stack Pointer. Además, posee tres parejas de registros de trabajo (B-C, D-E y H-L), así co-

mo un control de interrupciones y otro de entrada y salida de información en serie.

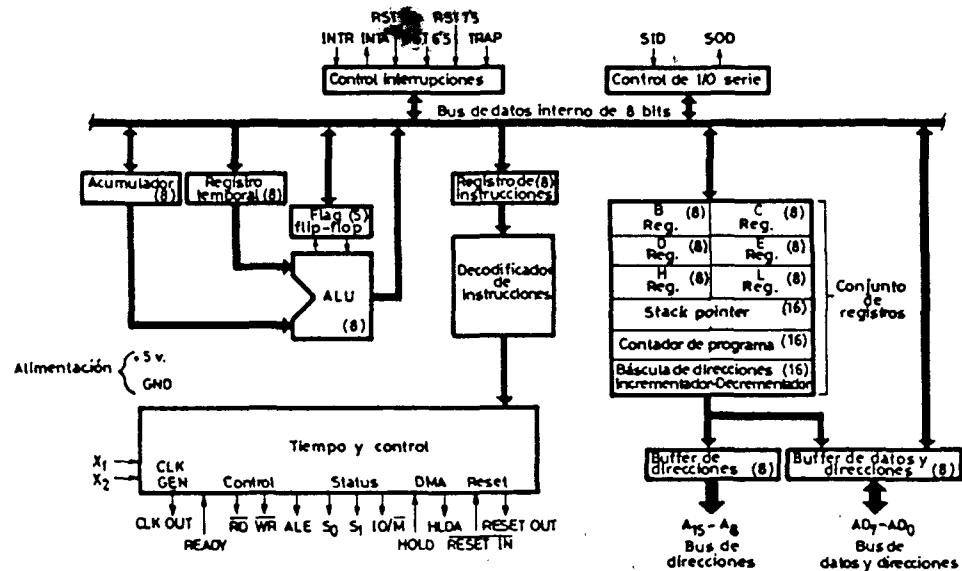


Fig. A-10.- Arquitectura fundamental del microprocesador 8085.

El generador de impulsos de reloj está integrado en el mismo chip y se encarga de proporcionar la señal de reloj (CLK OUT). Dicho generador precisa para la estabilización de su frecuencia de un cristal o red R-C aplicada a las entradas  $X_1$  y  $X_2$ .

El 8085 consta de tres bloques fundamentales.

- 1.º) Conjunto de registros, formado por el Contador de Programa (16 bits), el Stack Pointer (16) y las tres parejas de registros de 8 bits cada uno, junto con la báscula que incrementa o decrementa el contenido de todos estos registros.
- 2.º) La ALU, en combinación con el acumulador y un registro temporal, así como los 5 bits que actúan como flags o señalizadores de estado.
- 3.º) El Registro de Instrucciones, combinado con el decodificador de instrucciones y el circuito de control y tiempos.

El control de las interrupciones y el de la información de Entrada /Salida en serie se pueden considerar elementos auxiliares.



# 50

# Memoria y cartucho del MSX

## Introducción

La CPU del MSX, el microprocesador Z80, tiene un *bus* de direcciones de 16 bits, pudiendo direccionar hasta 64K bytes de memoria. La memoria de 64K está dividida en cuatro páginas de 16K. Puedes ampliar esta capacidad de memoria añadiendo un segmento a cada una de las páginas. Un segmento no es más que una memoria de 64K; la puedes considerar como una memoria aparte.

Físicamente un segmento es un cartucho con un banco de memoria cableada como la ROM del BASIC del MSX y su sistema de RAM.

Para fijar qué segmento de página se emplea, se encuentra el registro selector, en el cual habrás de introducir un valor. Lo veremos más adelante.

Generalmente los MSX tienen esta configuración de memoria:

PAGINA	SEGMENTO DEL SISTEMA	SEGMENTOS 1, 2 ó 3 CARTUCHO A CONECTAR
Página 3	RAM de 16K para los MSX de 16K	
Página 2	RAM de 16K para los MSX de 32K	ROM de 8 ó 16K para programas de juegos. Extensión de 16K para la RAM de MSX de 16K.
Página 1	ROM del BASIC del MSX	Empleada para una extensión del BASIC. ROM con el sistema operativo de diskette o para otros lenguajes.
Página 0	ROM de BASIC del MSX (BIOS)	

## Cartucho

Todos los MSX tienen al menos una ranura en la que podrás colocar tu cartucho, que tienen varias funciones. Aquí tienes una lista con algunas de ellas:

1. Extensión de la RAM. Generalmente para convertir las máquinas de 16K a 32K.
2. ROM con programas de juegos. El cartucho contendrá código máquina o un programa en BASIC.
3. ROM de expansión del BASIC. Conteniendo rutinas que amplían el BASIC del MSX. La instrucción CALL es la vía de acceso a estas rutinas. Algunos cartuchos tienen su propia RAM declarada como área de trabajo.
4. Cartucho de E/S. Para controlar una unidad de *floppy disc*, interfaz de impresora o lápiz óptico. Suele tener su propia ROM para las operaciones de E/S.

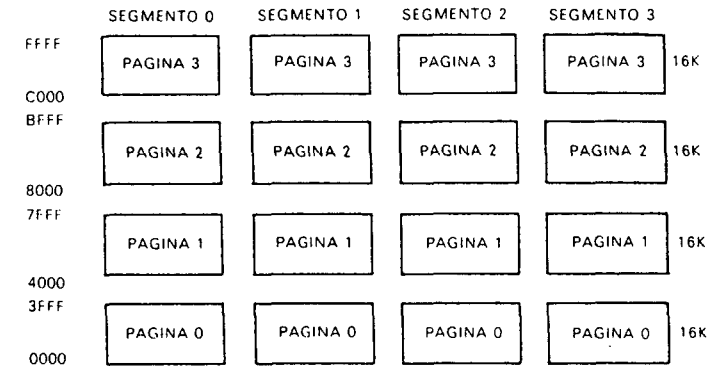
Puedes conectar cualquiera de estos cartuchos a cualquier ranura. El MSX tiene un mecanismo selector de cartucho, de modo que siempre sabrá a qué cartucho acceder.

## Organización de los segmentos básicos

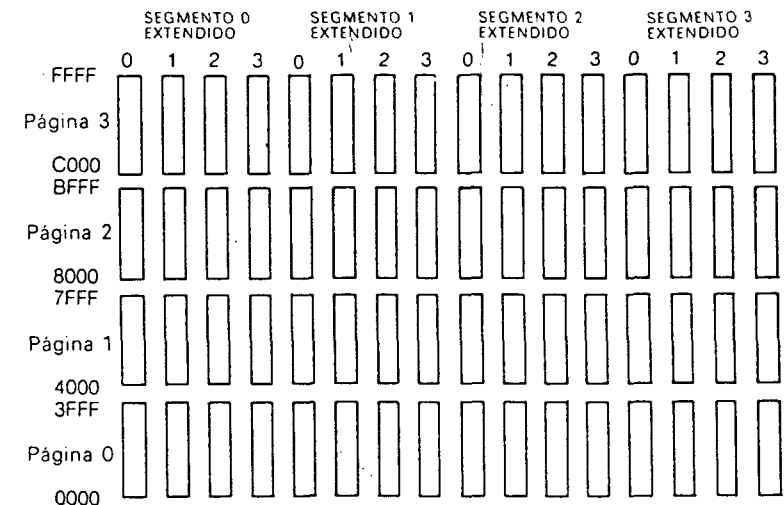
El MSX puede tener hasta cuatro segmentos; el segmento 0 es el que contiene la ROM con el BASIC del MSX (se le denomina segmento del sistema). A cada uno de estos cuatro segmentos se le puede conectar otros cuatro, teniendo así un total de dieciséis segmentos. Si cada uno de ellos tiene 64K de RAM, nos dan como resultado una memoria de 1M. Es la máxima memoria RAM que puede

llegar a direccionar el MSX. Memoria que no te es posible acceder desde el BASIC, ya que necesitas el MSX-DOS o programa en código máquina que te posibilite acceder a más de 64K de memoria.

### Organización de los segmentos



### Configuración de los segmentos de extensión



## Selector de segmento

Ya hemos dicho que el MSX puede tener varios segmentos, pero mientras no los controlemos éstos pueden interferir entre ellos.

Para determinar qué segmento se tomará para cada página, el MSX tiene un mecanismo especial de selección de registros.

En un determinado instante la CPU puede direccionar un máximo de 64K; o sea, cuatro páginas de RAM. Cada una de estas páginas puede ser de un segmento distinto. Las páginas de los segmentos se seleccionan mediante el puerto de salida A (8255 PPI) de 8 bits. (Véase el circuito selector de segmento al pie de página.)

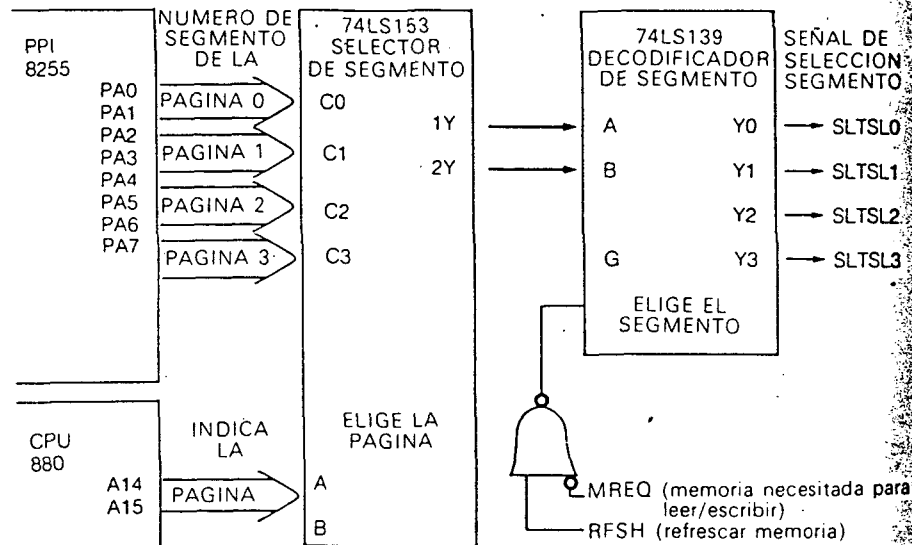
- PA0 y PA1 tienen el número de segmento para la página 0.
- PA2 y PA3 tienen el número de segmento para la página 1.
- PA4 y PA5 tienen el número de segmento para la página 2.
- PA6 y PA7 tienen el número de segmento para la página 3.

El PPI envía una señal al selector de segmento (74LS153). Este chip también recibe señales de la CPU (Z80) que le indican de qué página se está leyendo o escribiendo.

señal A15	señal A14	página de la que lee o escribe la CPU
0	0	página 0
0	1	página 1
1	0	página 2
1	1	página 3

Entonces se pasa el número de segmento seleccionado al decodificador 2 a 4 (74LS139), que envía la señal de selección de segmento, SLTSL, al segmento correspondiente de las cuatro páginas.

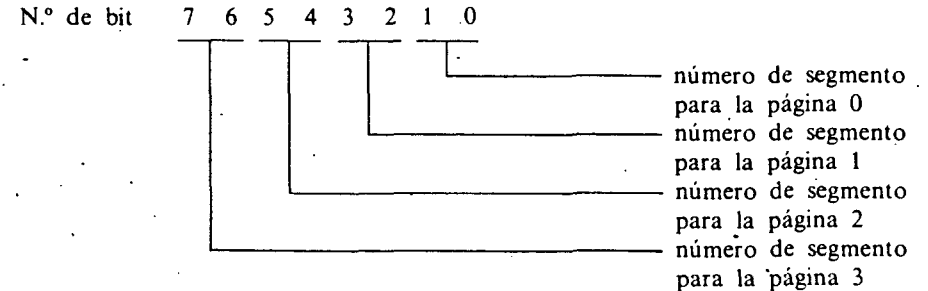
### DIAGRAMA DEL CIRCUITO SELECTOR DE SEGMENTO



## Cómo seleccionar y activar el segmento

Para seleccionar el segmento correspondiente a cada página por programa has de introducir un valor apropiado en la dirección &HA8, que es el puerto de salida A del PPI.

El valor a escribir es un número de 8 bits.



Ejemplo:

Supón que quieres las páginas 0, 1 y 3 del segmento 0 y la página 2 del segmento 1.

N.º de bit 7 6 5 4 3 2 1 0  
 0 0 0 1 0 0 0 0 = 16

Deberás escribir un 16 en la posición &HA8. Aunque la mejor manera de hacer esta operación es mediante BIOS, llamando a ENASLT (&H0024) desde el código máquina.

## Segmento de extensión

El MSX puede tener hasta cuatro segmentos, del 0 al 3. No todas las máquinas los tienen, aunque te ofrecen la posibilidad de conectarlos mediante los cartuchos. El número máximo de segmentos es de 16. Si cada uno de ellos es una RAM de 64K, el MSX puede tratar una memoria de 1M bytes.

Los segmentos de extensión no pueden tener, a su vez, extensiones; estos segmentos, cartuchos conectados, son extensiones de segmentos básicos y, por ello, sus cartuchos no llevan clavijas para conectar otros cartuchos.

Para seleccionar un segmento de extensión debemos seleccionar previamente el segmento básico al que está conectado, a través del puerto de salida A del PPI 8255. El registro de selección de segmento está en la posición &HFFFF del cartucho de extensión, e indica si la página seleccionada está a uno o no.

Para saber si un segmento básico tiene conectado algún segmento de extensión, usa POKE &HFFFF, &H0F. Ahora, PRINT PEEK (&HFFFF). Si el resultado es el complemento de lo que escribiste antes, entonces dicho segmento básico tiene conectado un segmento de extensión.

## Buffer del segmento de extensión

Los segmentos de extensión necesitan un *buffer*, ya que el cartucho con dicho segmento emplea un *bus* de dos direcciones.

El *buffer* cambia el sentido de transmisión según se lea o escriba. La señal de control BUSDIR es la que controla la dirección en que debe trabajar el *bus*.

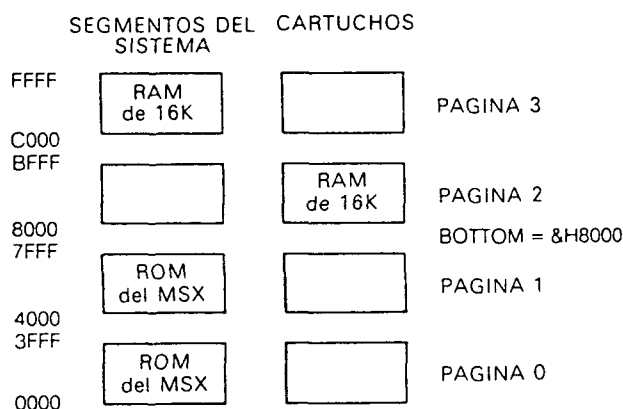
Las unidades que envían señales a la CPU, como los cartuchos de E/S, han de enviar la señal BUSDIR para cambiar el sentido de transmisión del *buffer* del segmento extendido con la CPU. Sin embargo, aquellos cartuchos que tengan sólo RAM o ROM no manejarán esta señal.

## Procedimiento de búsqueda de la RAM

Cuando conectas tu MSX lo primero que hace es localizar todos los segmentos para configurar la RAM del sistema:

1. Primero busca la RAM de la página 2, de la dirección &H8000 hasta la &HBFFF, y activa el segmento con mayor capacidad de RAM en esa página. Si hay más de un segmento con este máximo, se tomará el que tenga menor número de segmento.
2. Hace lo mismo con la página 3, de la dirección &HC000 a la &HFFFF, y toma el segmento con menor número y mayor capacidad de RAM.
3. Por último, analiza si la RAM es continua desde la dirección &H8000 a la &HFFFF, y asigna a la variable del sistema BOTTOM (&HFC48) la dirección de la posición con menor dirección disponible en la RAM.

### ORDENADOR DE 16K AMPLIADO A 32K



## Cartuchos de ROM con programas

### Procedimiento de búsqueda de la ROM

Después de seleccionar la RAM del sistema, el MSX localiza los cartuchos de ROM desde la dirección &H4000 a la &HBFFF, páginas 1 y 2. Analiza el área ID, al comienzo de cada página, desde el segmento 0 hasta el 3, además de los segmentos de extensión.

El área ID se encuentra al comienzo de cada página y tiene el siguiente formato:

+&H00	ID'AB'	ID...	Los dos bytes 'AB' indican que hay cartuchos de ROM.
+&H02	INI	INI...	Contiene la dirección de la rutina de inicialización de dicho cartucho.
+&H04	INSTRUC	INSTRUCCION...	Contiene la dirección de la rutina de tratamiento de las instrucciones ampliadas.
+&H06	UNIDAD	UNIDAD...	Contiene la dirección de la rutina de tratamiento de los periféricos ampliados.
+&H08	TEXTO	TEXTO...	Contiene la dirección de comienzo del programa BASIC en el cartucho.
+&H0A	reservado		
+&H10			

*Nota:* ID, INI, INSTRUCCION, UNIDAD y TEXTO contendrían ceros si no fuesen aplicables.

El BASIC del MSX realiza las siguientes operaciones para el procedimiento de localización del cartucho:

1. Analiza el contenido de ID y determina el tipo de rutina que contiene. Entonces pasa la información tomada al área de trabajo correspondiente.
2. Ejecuta la rutina INI, si la hubiere.
3. También ejecuta el programa BASIC, si éste existiese.

*Nota:* No se ejecutan ni INSTRUCCION ni UNIDAD, ya que éstas sólo se ejecutarán cuando necesites alguna instrucción ampliada o un periférico.

### INI: Rutina de inicialización

El contenido de INI es la dirección de la rutina de inicialización para dicho cartucho en concreto. Generalmente inicializa alguna unidad de E/S y fija el valor

de un vector (véase el tema de los vectores), o reserva un área de trabajo para el programa del cartucho.

La rutina de inicialización varía el contenido de todos los registros, excepto el puntero de pila [SP]. Cede el control al BASIC al encontrar una instrucción RET del Z80.

Esta rutina no tiene por qué ser de inicialización; puede ser un programa en código máquina que interesa ejecutar según conectes el ordenador. Puede ser incluso un programa de juegos.

De no haber rutina de inicialización, el contenido de INI serán ceros.

### TEXTO: Programa BASIC

Un cartucho de ROM no tiene por qué llevar sus programas escritos en código máquina: también pueden estar en BASIC. En TEXTO se halla la dirección de comienzo del programa BASIC que contiene el cartucho.

Cuando programes un cartucho en BASIC has de tener en cuenta lo siguiente:

1. Si tienes más de un cartucho con programas en BASIC conectado a tu ordenador, éste sólo ejecutará el que tenga menor número de segmento.
2. Los programas en BASIC de los cartuchos se almacenan en *tokens*.
3. Los cartuchos de este tipo han de estar asociados solamente a la página 2 (&H8000-&HBFFF). Por tanto, su capacidad máxima es de 16K.
4. La página 2 de cualquier otro segmento con RAM se desactivará y no se podrá emplear mientras se ejecute el cartucho con el programa en BASIC.
5. La dirección a la que apunta TEXTO ha de contener un cero.
6. Los números de línea de las instrucciones de salto, GOTO, GOSUB, etc. se constituirán por punteros para acelerar la ejecución.

En caso de que el cartucho no tenga un programa BASIC, TEXT estará a cero.

### INSTRUCCION: Rutina de ampliación de instrucciones

Mediante la instrucción CALL puedes llamar a ciertas instrucciones ampliadas que no estén en la ROM del BASIC del MSX. En INSTRUCCION está la dirección de comienzo de la rutina que ejecuta la primera instrucción ampliada del BASIC. El cartucho ha de estar en la página 1 de cualquier segmento, excepto en el segmento del sistema, que es el que contiene el BASIC.

La sintaxis de las instrucciones ampliadas es:

```
CALL <nombre-instrucción>  
CALL <nombre-instrucción> (argumentos)
```

La instrucción CALL se puede sustituir por un signo '-':

Cuando el BASIC encuentra una instrucción CALL, almacena el nombre de la instrucción extendida en el área de trabajo del sistema llamada PROCNM. Esta área comienza en la dirección &HFD89 y tiene 16 bytes de longitud. El nombre de la instrucción extendida se guarda en PROCNM seguido de un cero, por lo que las instrucciones no podrán tener más de quince caracteres.

Antes de ejecutar la instrucción ampliada, el puntero de texto, o sea, el registro [HL], contendrá la dirección que sigue a INSTRUCCION.

El procedimiento de INSTRUCCION analiza el contenido de PROCNM y ejecuta la rutina que corresponda a dicho nombre de instrucción.

Después de ejecutar la instrucción ampliada, el *flag* de acarreo se pondrá a 0 y el puntero de texto HL apuntará a la siguiente instrucción a ejecutar.

Si la instrucción ampliada almacenada en PROCNM no tuviese una rutina para ejecutarla, entonces el puntero de texto y el *flag* de acarreo quedarían como estaban y se enviará al BASIC un error sintáctico, mostrándote el mensaje *Syntax error* en pantalla.

En caso de que el cartucho no tuviese ninguna rutina de ejecución de instrucciones ampliadas, el campo INSTRUCCION estará a 0.

### UNIDAD: Rutina de tratamiento de extensiones de periféricos

El MSX te permite que le conectes cartuchos de extensión de E/S. El campo UNIDAD contendrá la dirección de comienzo de la rutina de tratamiento del periférico.

Recuerda estos puntos sobre las rutinas de UNIDAD:

1. El cartucho ha de estar conectado a la página 1 (&H4000-&HBFFF).
2. A un cartucho (16K) se le pueden conectar hasta cuatro unidades lógicas.
3. El nombre de la unidad se almacena en PROCNM, igual que se hacía en INSTRUCCION. PROCNM comienza en la dirección &HFD89 y tiene una longitud de 16 bytes. El nombre de la unidad va seguido de un 0 al guardarse en PROCNM; por tanto, la longitud máxima del nombre de una unidad es de dieciséis caracteres.
4. Cuando en una instrucción OPEN, u otras, el BASIC encuentra un nombre de unidad que no sea reconocida por la ROM del MSX, entonces hace una llamada a la unidad, metiendo &HFF en el acumulador. En caso de no existir una rutina que trate dicha unidad en el cartucho pone a 1 el *flag* de acarreo. Si ésta existe, el contenido de ID (valor de 0 a 3) se lleva al acumulador y el *flag* de acarreo se pone a 0. La rutina de dicha unidad puede trabajar con todos los registros.
5. Las operaciones de E/S se hacen realmente cuando se encuentre a UNIDAD con uno de estos valores en el acumulador:

- 0 Abrir (OPEN).
- 1 Cerrar (CLOSE).
- 4 E/S aleatoria.
- 6 Salida secuencial.
- 8 Entrada secuencial.
- 10 Función LOC.
- 12 Función LOF.
- 14 Función EOF.
- 16 Función FPOS.
- 18 Carácter de retroceso.

La variable del sistema UNIDAD ha de tener el contenido del indicador de unidad ID (0-3).

Si el cartucho no tuviera ninguna rutina de tratamiento de periféricos, entonces UNIDAD contendrá ceros.

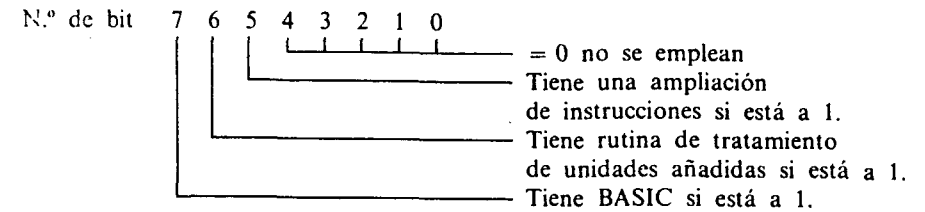
## Descripción de las variables del sistema relacionadas con el mecanismo de los segmentos

### Estado de los segmentos

EXPTBL:	Indica los segmentos ampliados. Está en la posición &HFCC1 y tiene cuatro bytes de longitud.
EXPTBL	&HFCC1 segmento 0. &HFCC2 segmento 1. &HFCC3 segmento 2. &HFCC4 segmento 3. &H80 indica segmento ampliado. &H00 no hay ampliación.
SLTTBL:	Indica el valor que se está sacando al registro selector de segmento de extensión. Sólo es válida en el caso de que el correspondiente EXPTBL tenga &H80; o sea, cuando se ha ampliado dicho segmento. Está en la posición &HFFC5 y tiene cuatro bytes de longitud.
SLTTBL	&HFFC5 segmento 0. &HFFC6 segmento 1. &HFFC7 segmento 2. &HFFC8 segmento 3.

### Estado de las páginas

SLTATR:	Indica el contenido de todas las páginas. Está en la posición &HFCC9 y tiene 64 bytes de longitud.
SLTATR	&HFCC9 segmento básico 0 segmento ampliado 0 página 0. &HFCCA segmento básico 0 segmento ampliado 0 página 1. ... ... &HFD07 segmento básico 3 segmento ampliado 3 página 2. &HFD08 segmento básico 3 segmento ampliado 3 página 3.



SLTWRK:	Area de trabajo de cada página. Su uso depende del contenido de la página, aunque cada página tiene dos bytes para indicar un área de trabajo, independientemente de su contenido. Está en la posición &HFD09 y tiene 128 bytes de longitud, 2 por página.
SLTWRK	&HFD09 segmento básico 0 segmento ampliado 0 página 0. &HFD0A segmento básico 0 segmento ampliado 0 página 0. &HFD0B segmento básico 0 segmento ampliado 0 página 1. &HFD0C segmento básico 0 segmento ampliado 0 página 1. ... ... &HFD85 segmento básico 3 segmento ampliado 3 página 2. &HFD86 segmento básico 3 segmento ampliado 3 página 2. &HFD87 segmento básico 3 segmento ampliado 3 página 3. &HFD88 segmento básico 3 segmento ampliado 3 página 3.

## 54/7493A 54LS/74LS93 DIVIDE-BY-SIXTEEN COUNTER

**DESCRIPTION** — The '93 is a 4-stage ripple counter containing a high speed flip-flop acting as a divide-by-two and three flip-flops connected as a divide-by-eight. HIGH signals on the Master Reset (MR) inputs override the clocks and force all outputs to the LOW state.

**ORDERING CODE:** See Section 9

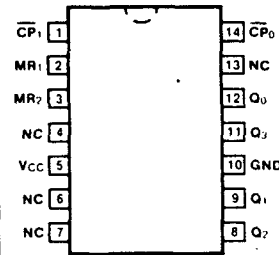
PKGS	PIN OUT	COMMERCIAL GRADE	MILITARY GRADE	PKG TYPE
		V <sub>CC</sub> = +5.0 V ±5%, T <sub>A</sub> = 0° C to +70° C	V <sub>CC</sub> = +5.0 V ±10%, T <sub>A</sub> = -55° C to +125° C	
Plastic DIP (P)	A	7493APC, 74LS93PC		9A
Ceramic DIP (D)	A	7493ADC, 74LS93DC	5493ADM, 54LS93DM	6A
Flatpak (F)	A	7493AFC, 74LS93FC	5493AFM, 54LS93FM	3I

**INPUT LOADING/FAN-OUT:** See Section 3 for U.L. definitions

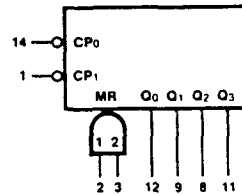
PIN NAMES	DESCRIPTION	54/74 (U.L.) HIGH/LOW	54/74LS (U.L.) HIGH/LOW
$\overline{CP}_0$	÷2 Section Clock Input (Active Falling Edge)	2.0/2.0	1.0/1.5
$\overline{CP}_1$	÷5 Section Clock Input (Active Falling Edge)	2.0/2.0	1.0/1.0
MR <sub>1</sub> , MR <sub>2</sub>	Asynchronous Master Reset Inputs (Active HIGH)	1.0/1.0	0.5/0.25
Q <sub>0</sub>	÷2 Section Output*	20/10	10/5.0 (2.5)
Q <sub>1</sub> — Q <sub>3</sub>	÷8 Section Outputs	20/10	10/5.0 (2.5)

\*The Q<sub>0</sub> output is guaranteed to drive the full rated fan-out plus the  $\overline{CP}_1$  input.

### CONNECTION DIAGRAM PINOUT A



### LOGIC SYMBOL



V<sub>CC</sub> = Pin 5  
GND = Pin 10  
NC = Pins 4, 6, 7, 13

**FUNCTIONAL DESCRIPTION** — The '93 is a 4-bit ripple type binary counter. It consists of four master/slave flip-flops which are internally connected to provide a divide-by-two section and a divide-by-eight section. Each section has a separate clock input which initiates state changes of the counter on the HIGH-to-LOW clock transition. State changes of the Q outputs do not occur simultaneously because of internal ripple delays. Therefore, decoded output signals are subject to decoding spikes and should not be used for clocks or strobes. The Q<sub>0</sub> output of each device is designed and specified to drive the rated fan-out plus the  $\overline{CP}_1$  input of the device. A gated AND asynchronous Master Reset (MR<sub>1</sub>, MR<sub>2</sub>) is provided which overrides the clocks and resets (clears) all the flip-flops. Since the output from the divide-by-two section is not internally connected to the succeeding stages, the devices may be operated in various counting modes.

- A. 4-Bit Ripple Counter — The output Q<sub>0</sub> must be externally connected to input  $\overline{CP}_1$ . The input count pulses are applied to input  $\overline{CP}_0$ . Simultaneous divisions of 2, 4, 8, and 16 are performed at the Q<sub>0</sub>, Q<sub>1</sub>, Q<sub>2</sub>, and Q<sub>3</sub> outputs as shown in the Truth Table.
- B. 3-Bit Ripple Counter — The input count pulses are applied to input  $\overline{CP}_1$ . Simultaneous frequency divisions of 2, 4, and 8 are available at the Q<sub>1</sub>, Q<sub>2</sub>, and Q<sub>3</sub> outputs. Independent use of the first flip-flop is available if the reset function coincides with reset of the 3-bit ripple-through counter.

### MODE SELECTION

RESET INPUTS		OUTPUTS			
MR <sub>1</sub>	MR <sub>2</sub>	Q <sub>0</sub>	Q <sub>1</sub>	Q <sub>2</sub>	Q <sub>3</sub>
H	H	L	L	L	L
L	H	Count			
H	L	Count			
L	L	Count			

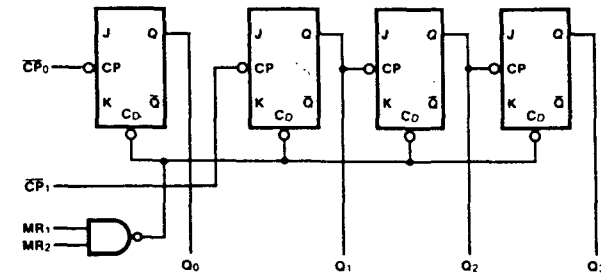
H = HIGH Voltage Level  
L = LOW Voltage Level

### TRUTH TABLE

COUNT	OUTPUTS			
	Q <sub>0</sub>	Q <sub>1</sub>	Q <sub>2</sub>	Q <sub>3</sub>
0	L	L	L	L
1	H	L	L	L
2	L	H	L	L
3	H	H	L	L
4	L	L	H	L
5	H	L	H	L
6	L	H	H	L
7	H	H	H	L
8	L	L	L	H
9	H	L	L	H
10	L	H	L	H
11	H	H	L	H
12	L	L	H	H
13	H	L	H	H
14	L	H	H	H
15	H	H	H	H

NOTE: Output Q<sub>0</sub> connected to  $\overline{CP}_1$ .

### LOGIC DIAGRAM





DC CHARACTERISTICS OVER OPERATING TEMPERATURE RANGE (unless otherwise specified)							
SYMBOL	PARAMETER	54/74		54/74LS		UNITS	CONDITIONS
		Min	Max	Min	Max		
I <sub>IH</sub>	Input HIGH Current CP <sub>0</sub> or CP <sub>1</sub>	1.0		0.2		mA	V <sub>CC</sub> = Max, V <sub>IN</sub> = 5.5 V
I <sub>CC</sub>	Power Supply Current	39		15		mA	V <sub>CC</sub> = Max

AC CHARACTERISTICS: V <sub>CC</sub> = +5.0 V, T <sub>A</sub> = +25° C (See Section 3 for waveforms and load configurations)							
SYMBOL	PARAMETER	54/74		54/74LS		UNITS	CONDITIONS
		C <sub>L</sub> = 15 pF R <sub>L</sub> = 400 Ω		C <sub>L</sub> = 15 pF			
		Min	Max	Min	Max		
f <sub>max</sub>	Maximum Count Frequency CP <sub>0</sub> Input	32		32		MHz	Figs. 3-1, 3-9
f <sub>max</sub>	Maximum Count Frequency CP <sub>1</sub> Input	16		16		MHz	Figs. 3-1, 3-9
t <sub>PLH</sub> t <sub>PHL</sub>	Propagation Delay CP <sub>0</sub> to Q <sub>0</sub>	16 18		16 18		ns	Figs. 3-1, 3-9
t <sub>PLH</sub> t <sub>PHL</sub>	Propagation Delay CP <sub>0</sub> to Q <sub>3</sub>	70 70		70 70		ns	Figs. 3-1, 3-9
t <sub>PLH</sub> t <sub>PHL</sub>	Propagation Delay CP <sub>1</sub> to Q <sub>1</sub>	16 21		16 21		ns	Figs. 3-1, 3-9
t <sub>PLH</sub> t <sub>PHL</sub>	Propagation Delay CP <sub>1</sub> to Q <sub>2</sub>	32 35		32 35		ns	Figs. 3-1, 3-9
t <sub>PLH</sub> t <sub>PHL</sub>	Propagation Delay CP <sub>1</sub> to Q <sub>3</sub>	51 51		51 51		ns	Figs. 3-1, 3-9
t <sub>PHL</sub>	Propagation Delay MR to Q <sub>n</sub>	40		40		ns	Figs. 3-1, 3-17

AC OPERATING REQUIREMENTS: V <sub>CC</sub> = +5.0 V, T <sub>A</sub> = +25° C							
SYMBOL	PARAMETER	54/74		54/74LS		UNITS	CONDITIONS
		Min	Max	Min	Max		
t <sub>w</sub> (H)	CP <sub>0</sub> Pulse Width HIGH	15		15		ns	Fig. 3-9
t <sub>w</sub> (H)	CP <sub>1</sub> Pulse Width HIGH	30		30		ns	Fig. 3-9
t <sub>w</sub> (H)	MR Pulse Width HIGH	15		15		ns	Fig. 3-17
t <sub>rec</sub>	Recovery Time, MR to CP	25		25		ns	Fig. 3-17

## 54/7494

### 4-BIT SHIFT REGISTER

**DESCRIPTION** — The '94 contains four dc coupled RS master/slave flip-flops with serial data entry into the first stage for synchronous Serial-in/Serial-out operation, and with a common asynchronous Clear and two sets of individual asynchronous Preset Inputs. Preset inputs P<sub>1X</sub> are enabled by a HIGH signal on PL<sub>1</sub> and Preset inputs P<sub>2X</sub> are enabled by a HIGH signal on PL<sub>2</sub>. The normal procedure for parallel entry of data consists of resetting the flip-flops by applying a momentary HIGH signal to CL, followed by a HIGH signal on either PL<sub>1</sub>, or PL<sub>2</sub>, depending on which set of parallel data is desired. For serial operation the CL and both PL inputs must be LOW. Serial transfer is initiated by the rising edge of the clock.

**ORDERING CODE:** See Section 9

PKGS	PIN OUT	COMMERCIAL GRADE	MILITARY GRADE	PKG TYPE
		V <sub>CC</sub> = +5.0 V ±5%, T <sub>A</sub> = 0° C to +70° C	V <sub>CC</sub> = +5.0 V ±10%, T <sub>A</sub> = -55° C to +125° C	
Plastic DIP (P)	A	7494PC		9B
Ceramic DIP (D)	A	7494DC	5494DM	7B
Flatpak (F)	A	7494FC	5494FM	4L

**CONNECTION DIAGRAM**  
PINOUT A

**INPUT LOADING/FAN-OUT:** See Section 3 for U.L. definitions

PIN NAMES	DESCRIPTION	54/74 (U.L.) HIGH/LOW
P <sub>1A</sub> — P <sub>1D</sub>	Source 1 Parallel Data Inputs	1.0/1.0
P <sub>2A</sub> — P <sub>2D</sub>	Source 2 Parallel Data Inputs	1.0/1.0
PL <sub>1</sub>	Asynchronous Parallel Load Input (Source 1)	4.0/4.0
PL <sub>2</sub>	Asynchronous Parallel Load Input (Source 2)	4.0/4.0
D <sub>s</sub>	Serial Data Input	1.0/1.0
CP	Clock Pulse Input (Active Rising Edge)	1.0/1.0
CL	Asynchronous Clear Input (Active HIGH)	1.0/1.0
Q <sub>0</sub>	Serial Data Output	10/10

**LOGIC SYMBOL**

V<sub>CC</sub> = Pin 5  
GND = Pin 12

# μA7800 SERIES

## 3-TERMINAL POSITIVE VOLTAGE REGULATORS

### FAIRCHILD LINEAR INTEGRATED CIRCUITS

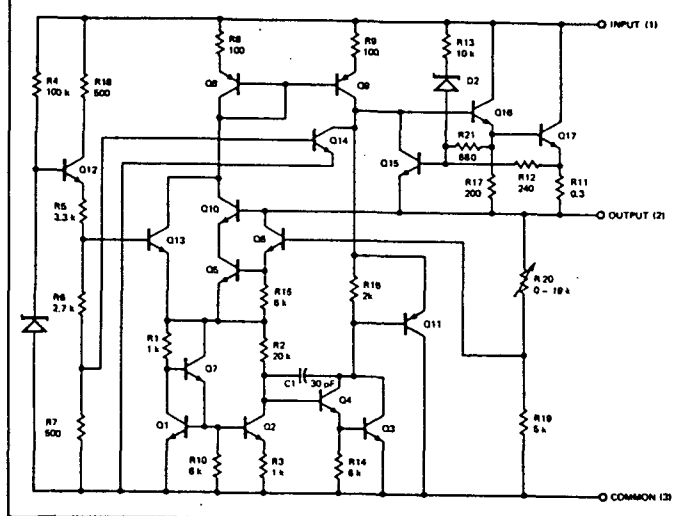
**GENERAL DESCRIPTION** — The μA7800 series of monolithic 3-Terminal Positive Voltage Regulators is constructed using the Fairchild Planar\* epitaxial process. These regulators employ internal current limiting, thermal shutdown and safe area compensation, making them essentially indestructible. If adequate heat sinking is provided, they can deliver over 1 A output current. They are intended as fixed voltage regulators in a wide range of applications including local (on card) regulation for elimination of distribution problems associated with single point regulation. In addition to use as fixed voltage regulators, these devices can be used with external components to obtain adjustable output voltages and currents.

- OUTPUT CURRENT IN EXCESS OF 1 A
- NO EXTERNAL COMPONENTS
- INTERNAL THERMAL OVERLOAD PROTECTION
- INTERNAL SHORT CIRCUIT CURRENT LIMITING
- OUTPUT TRANSISTOR SAFE AREA COMPENSATION
- AVAILABLE IN THE TO-220 AND THE TO-3 PACKAGE
- OUTPUT VOLTAGES OF 5, 6, 8, 8.5, 12, 15, 18, AND 24 V

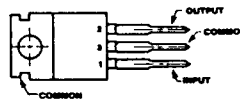
#### ABSOLUTE MAXIMUM RATINGS

Input Voltage (5 V through 18 V)	35 V
(24 V)	40 V
Internal Power Dissipation	Internally Limited
Storage Temperature Range	-65°C to +150°C
Operating Junction Temperature Range	-55°C to +150°C
	0°C to +150°C
Lead Temperature (Soldering, 60 s time limit) TO-3 Package	300°C
(Soldering, 10 s time limit) TO-220 Package	230°C

#### EQUIVALENT CIRCUIT



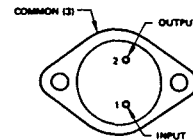
#### CONNECTION DIAGRAMS TO-220 PACKAGE (SIDE VIEW)



#### ORDER INFORMATION

OUTPUT VOLTAGE	TYPE	PART NO.
5 V	μA7805C	μA7805UC
6 V	μA7806C	μA7806UC
8 V	μA7808C	μA7808UC
8.5 V	μA7885C	μA7885UC
12 V	μA7812C	μA7812UC
15 V	μA7815C	μA7815UC
18 V	μA7818C	μA7818UC
24 V	μA7824C	μA7824UC

#### TO-3 PACKAGE (TOP VIEW)



#### ORDER INFORMATION

OUTPUT VOLTAGE	TYPE	PART NO.
5 V	μA7805	μA7805KM
6 V	μA7806	μA7806KM
8 V	μA7808	μA7808KM
8.5 V	μA7885	μA7885KM
12 V	μA7812	μA7812KM
15 V	μA7815	μA7815KM
18 V	μA7818	μA7818KM
24 V	μA7824	μA7824KM
5 V	μA7805C	μA7805KC
6 V	μA7806C	μA7806KC
8 V	μA7808C	μA7808KC
8.5 V	μA7885C	μA7885KC
12 V	μA7812C	μA7812KC
15 V	μA7815C	μA7815KC
18 V	μA7818C	μA7818KC
24 V	μA7824C	μA7824KC

\* Planar is a patented Fairchild process.

## FAIRCHILD • μA7800 SERIES

μA7805

ELECTRICAL CHARACTERISTICS:  $V_{IN} = 10\text{ V}$ ,  $I_{OUT} = 600\text{ mA}$ ,  $-55^\circ\text{C} < T_J < 150^\circ\text{C}$ ,  $C_{IN} = 0.33\ \mu\text{F}$ ,  $C_{OUT} = 0.1\ \mu\text{F}$ , unless otherwise specified.

CHARACTERISTICS	CONDITIONS (Note 1)	MIN	TYP	MAX	UNITS
Output Voltage	$T_J = 25^\circ\text{C}$	4.8	5.0	5.2	V
Line Regulation	$T_J = 25^\circ\text{C}$		3	50	mV
	$7\text{ V} < V_{IN} < 25\text{ V}$		1	25	mV
Load Regulation	$T_J = 25^\circ\text{C}$		15	100	mV
	$5\text{ mA} < I_{OUT} < 1.5\text{ A}$		5	25	mV
Output Voltage	$8.0\text{ V} < V_{IN} < 20\text{ V}$ $5\text{ mA} < I_{OUT} < 1.0\text{ A}$ $P < 15\text{ W}$	4.65		5.35	V
Quiescent Current	$T_J = 25^\circ\text{C}$		4.2	6.0	mA
Quiescent Current Change	with line			0.8	mA
	with load			0.5	mA
Output Noise Voltage	$T_A = 25^\circ\text{C}$ , $10\text{ Hz} < f < 100\text{ kHz}$		8	40	$\mu\text{V}/V_{OUT}$
Ripple Rejection	$f = 120\text{ Hz}$ , $8\text{ V} < V_{IN} < 18\text{ V}$	68	78		dB
Dropout Voltage	$I_{OUT} = 1.0\text{ A}$ , $T_J = 25^\circ\text{C}$		2.0	2.5	V
Output Resistance	$f = 1\text{ kHz}$		17		mΩ
Short Circuit Current	$T_J = 25^\circ\text{C}$ , $V_{IN} = 35\text{ V}$		0.75	1.2	A
Peak Output Current	$T_J = 25^\circ\text{C}$	1.3	2.2	3.3	A
Average Temperature Coefficient of Output Voltage	$I_{OUT} = 5\text{ mA}$			0.4	$\text{mV}/^\circ\text{C}$
	$-55^\circ\text{C} < T_J < +25^\circ\text{C}$			0.3	$\text{V}/\text{OUT}$

μA7805C

ELECTRICAL CHARACTERISTICS:  $V_{IN} = 10\text{ V}$ ,  $I_{OUT} = 600\text{ mA}$ ,  $0^\circ\text{C} < T_J < 125^\circ\text{C}$ ,  $C_{IN} = 0.33\ \mu\text{F}$ ,  $C_{OUT} = 0.1\ \mu\text{F}$ , unless otherwise specified.

CHARACTERISTICS	CONDITIONS (Note 1)	MIN	TYP	MAX	UNITS
Output Voltage	$T_J = 25^\circ\text{C}$	4.8	5.0	5.2	V
Line Regulation	$T_J = 25^\circ\text{C}$		3	100	mV
	$7\text{ V} < V_{IN} < 25\text{ V}$		1	50	mV
Load Regulation	$T_J = 25^\circ\text{C}$		15	100	mV
	$5\text{ mA} < I_{OUT} < 1.5\text{ A}$		5	50	mV
Output Voltage	$7\text{ V} < V_{IN} < 20\text{ V}$ $5\text{ mA} < I_{OUT} < 1.0\text{ A}$ $P < 15\text{ W}$	4.75		5.25	V
Quiescent Current	$T_J = 25^\circ\text{C}$		4.2	8.0	mA
Quiescent Current Change	with line			1.3	mA
	with load			0.5	mA
Output Noise Voltage	$T_A = 25^\circ\text{C}$ , $10\text{ Hz} < f < 100\text{ kHz}$		40		$\mu\text{V}$
Ripple Rejection	$f = 120\text{ Hz}$ , $8\text{ V} < V_{IN} < 18\text{ V}$	62	78		dB
Dropout Voltage	$I_{OUT} = 1.0\text{ A}$ , $T_J = 25^\circ\text{C}$		2.0		V
Output Resistance	$f = 1\text{ kHz}$		17		mΩ
Short Circuit Current	$T_J = 25^\circ\text{C}$ , $V_{IN} = 35\text{ V}$		750		mA
Peak Output Current	$T_J = 25^\circ\text{C}$		2.2		A
Average Temperature Coefficient of Output Voltage	$I_{OUT} = 5\text{ mA}$ , $0^\circ\text{C} < T_J < 125^\circ\text{C}$			-1.1	$\text{mV}/^\circ\text{C}$

#### NOTE:

1. All characteristics except noise voltage and ripple rejection ratio are measured using pulse techniques ( $t_w < 10\text{ ms}$ , duty cycle  $< 5\%$ ). Output voltage changes due to changes in internal temperature must be taken into account separately.

FAIRCHILD •  $\mu$ A7800 SERIES

$\mu$ A7806

ELECTRICAL CHARACTERISTICS:  $V_{IN} = 11\text{ V}$ ,  $I_{OUT} = 500\text{ mA}$ ,  $-55^\circ\text{C} < T_J < 150^\circ\text{C}$ ,  $C_{IN} = 0.33\ \mu\text{F}$ ,  $C_{OUT} = 0.1\ \mu\text{F}$ , unless otherwise specified.

CHARACTERISTICS	CONDITIONS (Note 1)	MIN	TYP	MAX	UNITS
Output Voltage	$T_J = 25^\circ\text{C}$	5.75	6.0	6.25	V
Line Regulation	$T_J = 25^\circ\text{C}$ , $8\text{ V} < V_{IN} < 25\text{ V}$		5	60	mV
	$9\text{ V} < V_{IN} < 13\text{ V}$		1.5	30	mV
Load Regulation	$T_J = 25^\circ\text{C}$ , $5\text{ mA} < I_{OUT} < 1.5\text{ A}$		14	100	mV
	$250\text{ mA} < I_{OUT} < 750\text{ mA}$		4	30	mV
Output Voltage	$9\text{ V} < V_{IN} < 21\text{ V}$ $5\text{ mA} < I_{OUT} < 1.0\text{ A}$ $P < 15\text{ W}$	5.65		6.35	V
Quiescent Current	$T_J = 25^\circ\text{C}$		4.3	6.0	mA
Quiescent Current Change	with line, $9\text{ V} < V_{IN} < 25\text{ V}$			0.8	mA
	with load, $5\text{ mA} < I_{OUT} < 1.0\text{ A}$			0.5	mA
Output Noise Voltage	$T_A = 25^\circ\text{C}$ , $10\text{ Hz} < f < 100\text{ kHz}$		8	40	$\mu\text{V}/V_{OUT}$
Ripple Rejection	$f = 120\text{ Hz}$ , $9\text{ V} < V_{IN} < 19\text{ V}$	65	75		dB
Dropout Voltage	$I_{OUT} = 1.0\text{ A}$ , $T_J = 25^\circ\text{C}$		2.0	2.5	V
Output Resistance	$f = 1\text{ kHz}$		19		m $\Omega$
Short Circuit Current	$T_J = 25^\circ\text{C}$ , $V_{IN} = 35\text{ V}$		0.75	1.2	A
Peak Output Current	$T_J = 25^\circ\text{C}$		2.2	3.3	A
Average Temperature Coefficient of Output Voltage	$I_{OUT} = 5\text{ mA}$ , $-55^\circ\text{C} < T_J < +25^\circ\text{C}$			0.4	mV/ $^\circ\text{C}$
	$+25^\circ\text{C} < T_J < 150^\circ\text{C}$			0.3	$V_{OUT}$

$\mu$ A7806C

ELECTRICAL CHARACTERISTICS:  $V_{IN} = 11\text{ V}$ ,  $I_{OUT} = 500\text{ mA}$ ,  $0^\circ\text{C} < T_J < 125^\circ\text{C}$ ,  $C_{IN} = 0.33\ \mu\text{F}$ ,  $C_{OUT} = 0.1\ \mu\text{F}$ , unless otherwise specified.

CHARACTERISTICS	CONDITIONS (Note 1)	MIN	TYP	MAX	UNITS
Output Voltage	$T_J = 25^\circ\text{C}$	5.75	6.0	6.25	V
Line Regulation	$T_J = 25^\circ\text{C}$ , $8\text{ V} < V_{IN} < 25\text{ V}$		5	120	mV
	$9\text{ V} < V_{IN} < 13\text{ V}$		1.5	60	mV
Load Regulation	$T_J = 25^\circ\text{C}$ , $5\text{ mA} < I_{OUT} < 1.5\text{ A}$		14	120	mV
	$250\text{ mA} < I_{OUT} < 750\text{ mA}$		4	60	mV
Output Voltage	$8\text{ V} < V_{IN} < 21\text{ V}$ $5\text{ mA} < I_{OUT} < 1.0\text{ A}$ $P < 15\text{ W}$	5.7		6.3	V
Quiescent Current	$T_J = 25^\circ\text{C}$		4.3	8.0	mA
Quiescent Current Change	with line, $8\text{ V} < V_{IN} < 25\text{ V}$			1.3	mA
	with load, $5\text{ mA} < I_{OUT} < 1.0\text{ A}$			0.5	mA
Output Noise Voltage	$T_A = 25^\circ\text{C}$ , $10\text{ Hz} < f < 100\text{ kHz}$		45		$\mu\text{V}$
Ripple Rejection	$f = 120\text{ Hz}$ , $9\text{ V} < V_{IN} < 19\text{ V}$	59	75		dB
Dropout Voltage	$I_{OUT} = 1.0\text{ A}$ , $T_J = 25^\circ\text{C}$		2.0		V
Output Resistance	$f = 1\text{ kHz}$		19		m $\Omega$
Short Circuit Current	$T_J = 25^\circ\text{C}$ , $V_{IN} = 35\text{ V}$		550		mA
Peak Output Current	$T_J = 25^\circ\text{C}$		2.2		A
Average Temperature Coefficient of Output Voltage	$I_{OUT} = 5\text{ mA}$ , $0^\circ\text{C} < T_J < 125^\circ\text{C}$		-0.8		mV/ $^\circ\text{C}$

NOTE: 1. All characteristics except noise voltage and ripple rejection ratio are measured using pulse techniques ( $t_w < 10\text{ ms}$ , duty cycle  $\leq 5\%$ ). Output voltage changes due to changes in internal temperature must be taken into account separately.

FAIRCHILD •  $\mu$ A7800 SERIES

$\mu$ A7808

ELECTRICAL CHARACTERISTICS:  $V_{IN} = 14\text{ V}$ ,  $I_{OUT} = 500\text{ mA}$ ,  $-55^\circ\text{C} < T_J < 150^\circ\text{C}$ ,  $C_{IN} = 0.33\ \mu\text{F}$ ,  $C_{OUT} = 0.1\ \mu\text{F}$ , unless otherwise specified.

CHARACTERISTICS	CONDITIONS (Note 1)	MIN	TYP	MAX	UNITS
Output Voltage	$T_J = 25^\circ\text{C}$	7.7	8.0	8.3	V
Line Regulation	$T_J = 25^\circ\text{C}$ , $10.5\text{ V} < V_{IN} < 25\text{ V}$		6.0	80	mV
	$11\text{ V} < V_{IN} < 17\text{ V}$		2.0	40	mV
Load Regulation	$T_J = 25^\circ\text{C}$ , $5\text{ mA} < I_{OUT} < 1.5\text{ A}$		12	100	mV
	$250\text{ mA} < I_{OUT} < 750\text{ mA}$		4.0	40	mV
Output Voltage	$11.5\text{ V} < V_{IN} < 23\text{ V}$ $5\text{ mA} < I_{OUT} < 1.0\text{ A}$ $P < 15\text{ W}$	7.6		8.4	V
Quiescent Current	$T_J = 25^\circ\text{C}$		4.3	6.0	mA
Quiescent Current Change	with line, $11.5\text{ V} < V_{IN} < 25\text{ V}$			0.8	mA
	with load, $5\text{ mA} < I_{OUT} < 1.0\text{ A}$			0.5	mA
Output Noise Voltage	$T_A = 25^\circ\text{C}$ , $10\text{ Hz} < f < 100\text{ kHz}$		8	40	$\mu\text{V}/V_{OUT}$
Ripple Rejection	$f = 120\text{ Hz}$ , $11.5\text{ V} < V_{IN} < 21.5\text{ V}$	62	72		dB
Dropout Voltage	$I_{OUT} = 1.0\text{ A}$ , $T_J = 25^\circ\text{C}$		2.0	2.5	V
Output Resistance	$f = 1\text{ kHz}$		18		m $\Omega$
Short Circuit Current	$T_J = 25^\circ\text{C}$ , $V_{IN} = 35\text{ V}$		0.75	1.2	A
Peak Output Current	$T_J = 25^\circ\text{C}$		1.3	2.2	A
Average Temperature Coefficient of Output Voltage	$I_{OUT} = 5\text{ mA}$ , $-55^\circ\text{C} < T_J < +25^\circ\text{C}$			0.4	mV/ $^\circ\text{C}$
	$+25^\circ\text{C} < T_J < 150^\circ\text{C}$			0.3	$V_{OUT}$

$\mu$ A7808C

ELECTRICAL CHARACTERISTICS:  $V_{IN} = 14\text{ V}$ ,  $I_{OUT} = 500\text{ mA}$ ,  $0^\circ\text{C} < T_J < 125^\circ\text{C}$ ,  $C_{IN} = 0.33\ \mu\text{F}$ ,  $C_{OUT} = 0.1\ \mu\text{F}$ , unless otherwise specified.

CHARACTERISTICS	CONDITIONS (Note 1)	MIN	TYP	MAX	UNITS
Output Voltage	$T_J = 25^\circ\text{C}$	7.7	8.0	8.3	V
Line Regulation	$T_J = 25^\circ\text{C}$ , $10.5\text{ V} < V_{IN} < 25\text{ V}$		6.0	160	mV
	$11\text{ V} < V_{IN} < 17\text{ V}$		2.0	80	mV
Load Regulation	$T_J = 25^\circ\text{C}$ , $5\text{ mA} < I_{OUT} < 1.5\text{ A}$		12	160	mV
	$250\text{ mA} < I_{OUT} < 750\text{ mA}$		4.0	80	mV
Output Voltage	$10.5\text{ V} < V_{IN} < 23\text{ V}$ $5\text{ mA} < I_{OUT} < 1.0\text{ A}$ $P < 15\text{ W}$	7.6		8.4	V
Quiescent Current	$T_J = 25^\circ\text{C}$		4.3	8.0	mA
Quiescent Current Change	with line, $10.5\text{ V} < V_{IN} < 25\text{ V}$			1.0	mA
	with load, $5\text{ mA} < I_{OUT} < 1.0\text{ A}$			0.5	mA
Output Noise Voltage	$T_A = 25^\circ\text{C}$ , $10\text{ Hz} < f < 100\text{ kHz}$		52		$\mu\text{V}$
Ripple Rejection	$f = 120\text{ Hz}$ , $11.5\text{ V} < V_{IN} < 21.5\text{ V}$	56	72		dB
Dropout Voltage	$I_{OUT} = 1.0\text{ A}$ , $T_J = 25^\circ\text{C}$		2.0		V
Output Resistance	$f = 1\text{ kHz}$		18		m $\Omega$
Short Circuit Current	$T_J = 25^\circ\text{C}$ , $V_{IN} = 35\text{ V}$		450		mA
Peak Output Current	$T_J = 25^\circ\text{C}$		2.2		A
Average Temperature Coefficient of Output Voltage	$I_{OUT} = 5\text{ mA}$ , $0^\circ\text{C} < T_J < 125^\circ\text{C}$		-0.8		mV/ $^\circ\text{C}$

NOTE: 1. All characteristics except noise voltage and ripple rejection ratio are measured using pulse techniques ( $t_w < 10\text{ ms}$ , duty cycle  $\leq 5\%$ ). Output voltage changes due to changes in internal temperature must be taken into account separately.

7

© Del documento, los autores. Digitalización realizada por ULPGC. Biblioteca Universitaria, 2008

**FAIRCHILD •  $\mu$ A7800 SERIES**

**$\mu$ A7885**

ELECTRICAL CHARACTERISTICS:  $V_{IN} = 15$  V,  $I_{OUT} = 500$  mA,  $0^\circ\text{C} < T_J < 125^\circ\text{C}$ ,  $C_{IN} = 0.33$   $\mu$ F,  $C_{OUT} = 0.1$   $\mu$ F, unless otherwise specified.

CHARACTERISTICS	CONDITIONS (Note 1)	MIN	TYP	MAX	UNITS
Output Voltage	$T_J = 25^\circ\text{C}$	8.15	8.5	8.85	V
Line Regulation	$T_J = 25^\circ\text{C}$	$10.5\text{ V} < V_{IN} < 25\text{ V}$	6.0	85	mV
		$11\text{ V} < V_{IN} < 17\text{ V}$	2.0	40	mV
Load Regulation	$T_J = 25^\circ\text{C}$	$5\text{ mA} < I_{OUT} < 1.5\text{ A}$	12	85	mV
		$250\text{ mA} < I_{OUT} < 750\text{ mA}$	4.0	40	mV
Output Voltage	$12\text{ V} < V_{IN} < 23.5\text{ V}$ $5\text{ mA} < I_{OUT} < 1.0\text{ A}$ $P < 15\text{ W}$	8.1		8.9	V
Quiescent Current	$T_J = 25^\circ\text{C}$		4.3	6.0	mA
Quiescent Current Change	with line	$11.5\text{ V} < V_{IN} < 25\text{ V}$		0.8	mA
		$5\text{ mA} < I_{OUT} < 1.0\text{ A}$		0.5	mA
Output Noise Voltage	$T_A = 25^\circ\text{C}$ , $10\text{ Hz} < f < 100\text{ kHz}$		8	40	$\mu\text{V}/V_{OUT}$
Ripple Rejection	$f = 120\text{ Hz}$ , $11.5\text{ V} < V_{IN} < 21.5\text{ V}$	62	70		dB
Dropout Voltage	$I_{OUT} = 1.0\text{ A}$ , $T_J = 25^\circ\text{C}$		2.0	2.5	V
Output Resistance	$f = 1\text{ kHz}$		16		m $\Omega$
Short Circuit Current	$T_J = 25^\circ\text{C}$ , $V_{IN} = 35\text{ V}$		0.75	1.2	A
Peak Output Current	$T_J = 25^\circ\text{C}$	1.3	2.2	3.3	A
Average Temperature Coefficient of Output Voltage	$I_{OUT} = 5\text{ mA}$	$-55^\circ\text{C} < T_J < +25^\circ\text{C}$		0.4	$\text{mV}/^\circ\text{C}$
		$+25^\circ\text{C} < T_J < +150^\circ\text{C}$		0.3	$V_{OUT}$

**$\mu$ A7885C**

ELECTRICAL CHARACTERISTICS:  $V_{IN} = 15$  V,  $I_{OUT} = 500$  mA,  $-55^\circ\text{C} < T_J < 150^\circ\text{C}$ ,  $C_{IN} = 0.33$   $\mu$ F,  $C_{OUT} = 0.1$   $\mu$ F, unless otherwise specified.

CHARACTERISTICS	CONDITIONS (Note 1)	MIN	TYP	MAX	UNITS
Output Voltage	$T_J = 25^\circ\text{C}$	8.15	8.5	8.85	V
Line Regulation	$T_J = 25^\circ\text{C}$	$10.5\text{ V} < V_{IN} < 25\text{ V}$	6.0	170	mV
		$11\text{ V} < V_{IN} < 17\text{ V}$	2.0	85	mV
Load Regulation	$T_J = 25^\circ\text{C}$	$5\text{ mA} < I_{OUT} < 1.5\text{ A}$	12	170	mV
		$250\text{ mA} < I_{OUT} < 750\text{ mA}$	4.0	85	mV
Output Voltage	$11\text{ V} < V_{IN} < 23.5\text{ V}$ $5\text{ mA} < I_{OUT} < 1.0\text{ A}$ $P < 15\text{ W}$	8.1		8.9	V
Quiescent Current	$T_J = 25^\circ\text{C}$		4.3	8.0	mA
Quiescent Current Change	with line	$10.5\text{ V} < V_{IN} < 25\text{ V}$		1.0	mA
		$5\text{ mA} < I_{OUT} < 1.0\text{ A}$		0.5	mA
Output Noise Voltage	$T_A = 25^\circ\text{C}$ , $10\text{ Hz} < f < 100\text{ kHz}$		55		$\mu\text{V}$
Ripple Rejection	$f = 120\text{ Hz}$ , $11.5\text{ V} < V_{IN} < 21.5\text{ V}$	56	70		dB
Dropout Voltage	$I_{OUT} = 1.0\text{ A}$ , $T_J = 25^\circ\text{C}$		2.0		V
Output Resistance	$f = 1\text{ kHz}$		16		m $\Omega$
Short Circuit Current	$T_J = 25^\circ\text{C}$ , $V_{IN} = 35\text{ V}$		450		mA
Peak Output Current	$T_J = 25^\circ\text{C}$		2.2		A
Average Temperature Coefficient of Output Voltage	$I_{OUT} = 5\text{ mA}$ , $0^\circ\text{C} < T_J < 125^\circ\text{C}$		-0.8		$\text{mV}/^\circ\text{C}$

**NOTE:**

1. All characteristics except noise voltage and ripple rejection ratio are measured using pulse techniques ( $t_w < 10$  ms, duty cycle  $< 5\%$ ). Output voltage changes due to changes in internal temperature must be taken into account separately.

**FAIRCHILD •  $\mu$ A7800 SERIES**

**$\mu$ A7812**

ELECTRICAL CHARACTERISTICS:  $V_{IN} = 19$  V,  $I_{OUT} = 500$  mA,  $-55^\circ\text{C} < T_J < 150^\circ\text{C}$ ,  $C_{IN} = 0.33$   $\mu$ F,  $C_{OUT} = 0.1$   $\mu$ F, unless otherwise specified.

CHARACTERISTICS	CONDITIONS (Note 1)	MIN	TYP	MAX	UNITS	
Output Voltage	$T_J = 25^\circ\text{C}$	11.5	12.0	12.5	V	
Line Regulation	$T_J = 25^\circ\text{C}$	$14.5\text{ V} < V_{IN} < 30\text{ V}$		10	120	mV
		$18\text{ V} < V_{IN} < 22\text{ V}$		3.0	60	mV
Load Regulation	$T_J = 25^\circ\text{C}$	$5\text{ mA} < I_{OUT} < 1.5\text{ A}$		12	120	mV
		$250\text{ mA} < I_{OUT} < 750\text{ mA}$		4.0	60	mV
Output Voltage	$15.5\text{ V} < V_{IN} < 27\text{ V}$ $5\text{ mA} < I_{OUT} < 1.0\text{ A}$ $P < 15\text{ W}$	11.4		12.6	V	
Quiescent Current	$T_J = 25^\circ\text{C}$		4.3	6.0	mA	
Quiescent Current Change	with line	$15\text{ V} < V_{IN} < 30\text{ V}$		0.8	mA	
		$5\text{ mA} < I_{OUT} < 1.0\text{ A}$		0.5	mA	
Output Noise Voltage	$T_A = 25^\circ\text{C}$ , $10\text{ Hz} < f < 100\text{ kHz}$		8	40	$\mu\text{V}/V_{OUT}$	
Ripple Rejection	$f = 120\text{ Hz}$ , $15\text{ V} < V_{IN} < 25\text{ V}$	61	71		dB	
Dropout Voltage	$I_{OUT} = 1.0\text{ A}$ , $T_J = 25^\circ\text{C}$		2.0	2.5	V	
Output Resistance	$f = 1\text{ kHz}$		18		m $\Omega$	
Short Circuit Current	$T_J = 25^\circ\text{C}$ , $V_{IN} = 35\text{ V}$		0.75	1.2	A	
Peak Output Current	$T_J = 25^\circ\text{C}$	1.3	2.2	3.3	A	
Average Temperature Coefficient of Output Voltage	$I_{OUT} = 5\text{ mA}$	$-55^\circ\text{C} < T_J < +25^\circ\text{C}$		0.4	$\text{mV}/^\circ\text{C}$	
		$+25^\circ\text{C} < T_J < +150^\circ\text{C}$		0.3	$V_{OUT}$	

**$\mu$ A7812C**

ELECTRICAL CHARACTERISTICS:  $V_{IN} = 19$  V,  $I_{OUT} = 500$  mA,  $0^\circ\text{C} < T_J < 125^\circ\text{C}$ ,  $C_{IN} = 0.33$   $\mu$ F,  $C_{OUT} = 0.1$   $\mu$ F, unless otherwise specified.

CHARACTERISTICS	CONDITIONS (Note 1)	MIN	TYP	MAX	UNITS	
Output Voltage	$T_J = 25^\circ\text{C}$	11.5	12.0	12.5	V	
Line Regulation	$T_J = 25^\circ\text{C}$	$14.5\text{ V} < V_{IN} < 30\text{ V}$		10	240	mV
		$18\text{ V} < V_{IN} < 22\text{ V}$		3.0	120	mV
Load Regulation	$T_J = 25^\circ\text{C}$	$5\text{ mA} < I_{OUT} < 1.5\text{ A}$		12	240	mV
		$250\text{ mA} < I_{OUT} < 750\text{ mA}$		4.0	120	mV
Output Voltage	$14.5\text{ V} < V_{IN} < 27\text{ V}$ $5\text{ mA} < I_{OUT} < 1.0\text{ A}$ $P < 15\text{ W}$	11.4		12.6	V	
Quiescent Current	$T_J = 25^\circ\text{C}$		4.3	8.0	mA	
Quiescent Current Change	with line	$14.5\text{ V} < V_{IN} < 30\text{ V}$		1.0	mA	
		$5\text{ mA} < I_{OUT} < 1.0\text{ A}$		0.5	mA	
Output Noise Voltage	$T_A = 25^\circ\text{C}$ , $10\text{ Hz} < f < 100\text{ kHz}$		75		$\mu\text{V}$	
Ripple Rejection	$f = 120\text{ Hz}$ , $15\text{ V} < V_{IN} < 25\text{ V}$	55	71		dB	
Dropout Voltage	$I_{OUT} = 1.0\text{ A}$ , $T_J = 25^\circ\text{C}$		2.0		V	
Output Resistance	$f = 1\text{ kHz}$		18		m $\Omega$	
Short Circuit Current	$T_J = 25^\circ\text{C}$ , $V_{IN} = 35\text{ V}$		350		mA	
Peak Output Current	$T_J = 25^\circ\text{C}$		2.2		A	
Average Temperature Coefficient of Output Voltage	$I_{OUT} = 5\text{ mA}$ , $0^\circ\text{C} < T_J < 125^\circ\text{C}$		-1.0		$\text{mV}/^\circ\text{C}$	

**NOTE:**

1. All characteristics except noise voltage and ripple rejection ratio are measured using pulse techniques ( $t_w < 10$  ms, duty cycle  $< 5\%$ ). Output voltage changes due to changes in internal temperature must be taken into account separately.

FAIRCHILD •  $\mu$ A7800 SERIES

$\mu$ A7824

ELECTRICAL CHARACTERISTICS:  $V_{IN} = 33$  V,  $I_{OUT} = 500$  mA,  $-55^{\circ}\text{C} < T_J < 150^{\circ}\text{C}$ ,  $C_{IN} = 0.33$   $\mu$ F,  $C_{OUT} = 0.1$   $\mu$ F, unless otherwise specified.

CHARACTERISTICS	CONDITIONS (Note 1)	MIN	TYP	MAX	UNITS
Output Voltage	$T_J = 25^{\circ}\text{C}$	23.0	24.0	25.0	V
Line Regulation	$T_J = 25^{\circ}\text{C}$		18	240	mV
	$27\text{ V} < V_{IN} < 38\text{ V}$ $30\text{ V} < V_{IN} < 36\text{ V}$		6	120	mV
Load Regulation	$T_J = 25^{\circ}\text{C}$		12	240	mV
	$5\text{ mA} < I_{OUT} < 1.5\text{ A}$ $250\text{ mA} < I_{OUT} < 750\text{ mA}$		4	120	mV
Output Voltage	$28\text{ V} < V_{IN} < 38\text{ V}$ $5\text{ mA} < I_{OUT} < 1.0\text{ A}$ $P < 15\text{ W}$	22.8		25.2	V
Quiescent Current	$T_J = 25^{\circ}\text{C}$		4.6	6.0	mA
Quiescent Current Change	with line			0.8	mA
	with load			0.5	mA
Output Noise Voltage	$T_A = 25^{\circ}\text{C}$ , $10\text{ Hz} < f < 100\text{ kHz}$		8	40	$\mu\text{V}/V_{OUT}$
Ripple Rejection	$f = 120\text{ Hz}$ , $28\text{ V} < V_{IN} < 38\text{ V}$	56	66		dB
Dropout Voltage	$I_{OUT} = 1.0\text{ A}$ , $T_J = 25^{\circ}\text{C}$		2.0	2.5	V
Output Resistance	$f = 1\text{ kHz}$		28		m $\Omega$
Short Circuit Current	$T_J = 25^{\circ}\text{C}$ , $V_{IN} = 35\text{ V}$		0.75	1.2	A
Peak Output Current	$T_J = 25^{\circ}\text{C}$	1.3	2.2	3.3	A
Average Temperature Coefficient of Output Voltage	$I_{OUT} = 5\text{ mA}$	$-55^{\circ}\text{C} < T_J < +25^{\circ}\text{C}$		0.4	mV/ $^{\circ}\text{C}$
		$+25^{\circ}\text{C} < T_J < +150^{\circ}\text{C}$		0.3	V/OUT

$\mu$ A7824C

ELECTRICAL CHARACTERISTICS:  $V_{IN} = 33$  V,  $I_{OUT} = 500$  mA,  $0^{\circ}\text{C} < T_J < 125^{\circ}\text{C}$ ,  $C_{IN} = 0.33$   $\mu$ F,  $C_{OUT} = 0.1$   $\mu$ F, unless otherwise specified.

CHARACTERISTICS	CONDITIONS (Note 1)	MIN	TYP	MAX	UNITS
Output Voltage	$T_J = 25^{\circ}\text{C}$	23.0	24.0	25.0	V
Line Regulation	$T_J = 25^{\circ}\text{C}$		18	480	mV
	$27\text{ V} < V_{IN} < 38\text{ V}$ $30\text{ V} < V_{IN} < 36\text{ V}$		6	240	mV
Load Regulation	$T_J = 25^{\circ}\text{C}$		12	480	mV
	$5\text{ mA} < I_{OUT} < 1.5\text{ A}$ $250\text{ mA} < I_{OUT} < 750\text{ mA}$		4	240	mV
Output Voltage	$27\text{ V} < V_{IN} < 38\text{ V}$ $5\text{ mA} < I_{OUT} < 1.0\text{ A}$ $P < 15\text{ W}$	22.8		25.2	V
Quiescent Current	$T_J = 25^{\circ}\text{C}$		4.6	8.0	mA
Quiescent Current Change	with line			1.0	mA
	with load			0.5	mA
Output Noise Voltage	$T_A = 25^{\circ}\text{C}$ , $10\text{ Hz} < f < 100\text{ kHz}$		170		$\mu\text{V}$
Ripple Rejection	$f = 120\text{ Hz}$ , $28\text{ V} < V_{IN} < 38\text{ V}$	50	66		dB
Dropout Voltage	$I_{OUT} = 1.0\text{ A}$ , $T_J = 25^{\circ}\text{C}$		2.0		V
Output Resistance	$f = 1\text{ kHz}$		28		m $\Omega$
Short Circuit Current	$T_J = 25^{\circ}\text{C}$ , $V_{IN} = 35\text{ V}$		150		mA
Peak Output Current	$T_J = 25^{\circ}\text{C}$		2.1		A
Average Temperature Coefficient of Output Voltage	$I_{OUT} = 5\text{ mA}$ , $0^{\circ}\text{C} < T_J < 125^{\circ}\text{C}$		-1.5		mV/ $^{\circ}\text{C}$

NOTE: 1. All characteristics except noise voltage and ripple rejection ratio are measured using pulse techniques ( $t_w < 10$  ms, duty cycle  $< 5\%$ ). Output voltage changes due to changes in internal temperature must be taken into account separately.

FAIRCHILD •  $\mu$ A7800 SERIES

DESIGN CONSIDERATIONS

The  $\mu$ A7800 fixed voltage regulator series has thermal overload protection from excessive power, internal short circuit protection which limits the regulator's maximum current, and output transistor safe area compensation for reducing the output current as the voltage across the pass transistor is increased.

Although the internal power dissipation is limited, the junction temperature must be kept below the maximum specified temperature ( $150^{\circ}\text{C}$  for 7800,  $125^{\circ}\text{C}$  for 7800C) in order to meet data sheet specifications. To calculate the maximum junction temperature or heat sink required, the following thermal resistance values should be used:

Package	Typ $\theta_{JC}$	Max $\theta_{JC}$	Typ $\theta_{JA}$	Max $\theta_{JA}$
TO-3	3.5	5.5	40	45
TO-220	3.0	5.0	60	65

$$P_D(\text{MAX}) = \frac{T_J(\text{MAX}) - T_A}{\theta_{JC} + \theta_{CA}} \quad \text{or} \quad \frac{T_J(\text{MAX}) - T_A}{\theta_{JA}} \quad (\text{Without a heat sink})$$

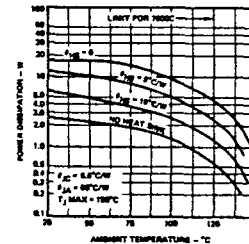
$$\theta_{CA} = \theta_{CS} + \theta_{SA}$$

Solving for  $T_J$ :  $T_J = T_A + P_D(\theta_{JC} + \theta_{CA})$  or  $T_A + P_D\theta_{JA}$  (without a heat sink)

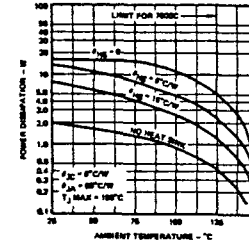
where  $T_J$  = Junction Temperature  
 $T_A$  = Ambient Temperature  
 $P_D$  = Power Dissipation

$\theta_{JC}$  = Junction to case thermal resistance  
 $\theta_{CA}$  = Case to ambient thermal resistance  
 $\theta_{CS}$  = Case to heat sink to resistance  
 $\theta_{SA}$  = Heat sink to ambient thermal resistance  
 $\theta_{JA}$  = Junction to ambient thermal resistance

WORST CASE POWER DISSIPATION VERSUS AMBIENT TEMPERATURE (TO-3)

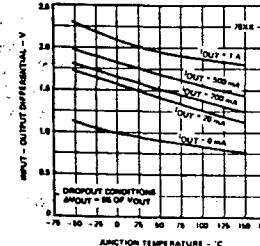


WORST CASE POWER DISSIPATION VERSUS AMBIENT TEMPERATURE (TO-220)

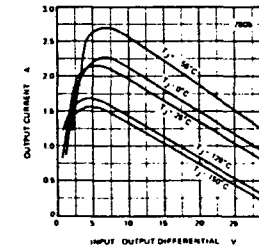


TYPICAL PERFORMANCE CURVES

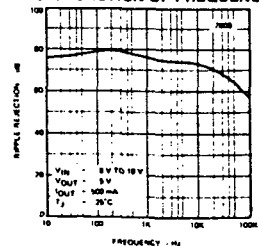
DROPOUT VOLTAGE AS A FUNCTION OF JUNCTION TEMPERATURE



PEAK OUTPUT CURRENT AS A FUNCTION OF INPUT/OUTPUT DIFFERENTIAL VOLTAGE

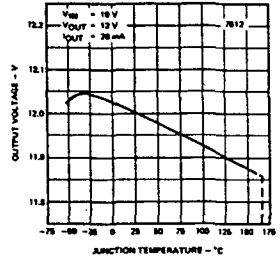


RIPLLE REJECTION AS A FUNCTION OF FREQUENCY

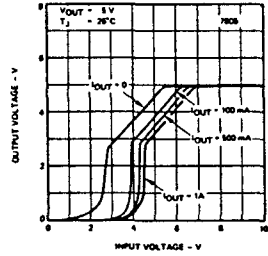


TYPICAL PERFORMANCE CURVES (Cont'd)

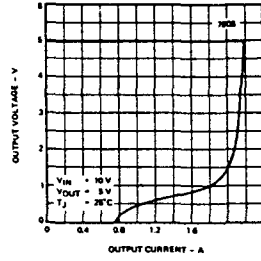
OUTPUT VOLTAGE AS A FUNCTION OF JUNCTION TEMPERATURE



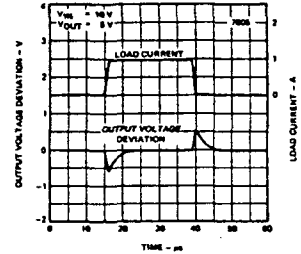
DROPOUT CHARACTERISTICS



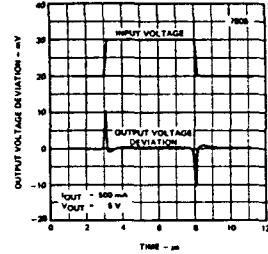
CURRENT LIMITING CHARACTERISTICS



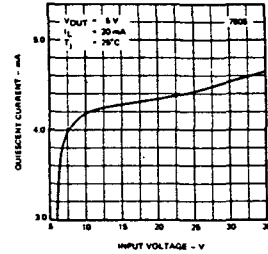
LOAD TRANSIENT RESPONSE



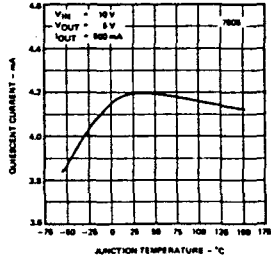
LINE TRANSIENT RESPONSE



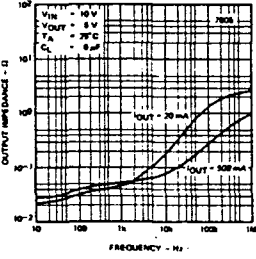
QUIESCENT CURRENT AS A FUNCTION OF INPUT VOLTAGE



QUIESCENT CURRENT AS A FUNCTION OF TEMPERATURE

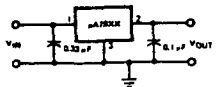


OUTPUT IMPEDANCE AS A FUNCTION OF FREQUENCY

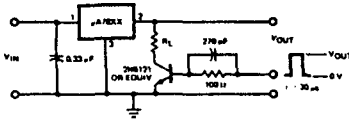


Note: The other  $\mu$ A7800 series devices have similar curves.

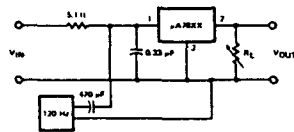
EQUIVALENT TEST CIRCUITS



IC PARAMETER TEST CIRCUIT

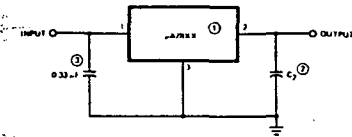


LOAD REGULATION TEST CIRCUIT

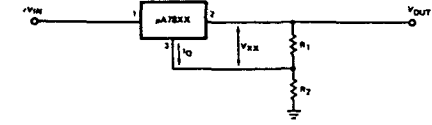


RIPPLE REJECTION TEST CIRCUIT

TYPICAL APPLICATIONS



FIXED OUTPUT REGULATOR

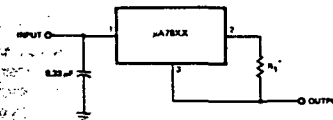


CIRCUIT FOR INCREASING OUTPUT VOLTAGE

NOTES:

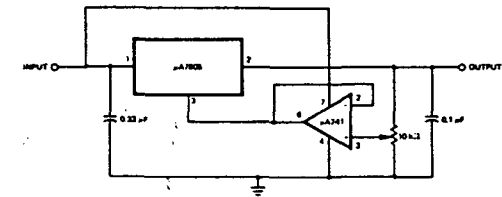
- ① To specify an output voltage, substitute voltage value for "XX".
- ② Although no output capacitor is needed for stability, it does improve transient response.
- ③ Required if regulator is located an appreciable distance from power supply filter.

$$V_{OUT} = V_{XX} \left(1 + \frac{R_2}{R_1}\right) + I_{QR2}$$

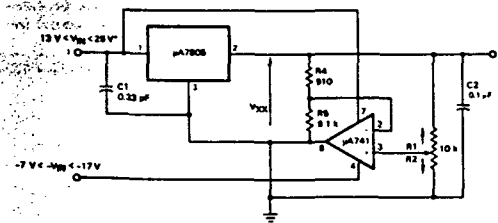


$$\text{Output Current} = \frac{V_{OUT}}{R_1}$$

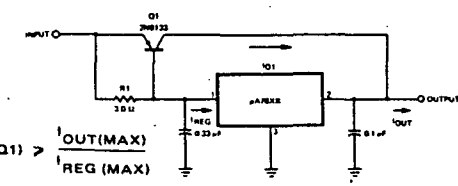
CURRENT REGULATOR



ADJUSTABLE OUTPUT REGULATOR, 7 TO 30 VOLTS



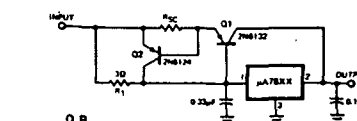
0.5 TO 10 V REGULATOR



$$\beta(Q1) > \frac{I_{OUT(MAX)}}{I_{REG(MAX)}}$$

$$R_1 = \frac{0.9}{I_{REG}} = \frac{\beta(Q1) V_{BE(Q1)}}{I_{REG(MAX)} (\beta + 1) - I_{OUT(MAX)}}$$

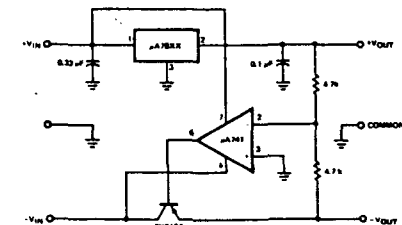
HIGH CURRENT VOLTAGE REGULATOR



$$R_{SC} = \frac{0.8}{I_{SC}}$$

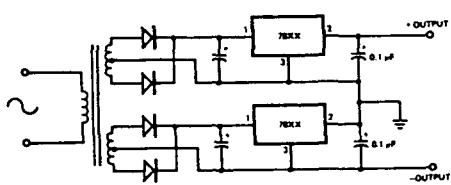
$$R_1 = \frac{\beta V_{BE(Q1)}}{I_{REG(MAX)} (\beta + 1) - I_{OUT(MAX)}}$$

HIGH OUTPUT CURRENT, SHORT CIRCUIT PROTECTED

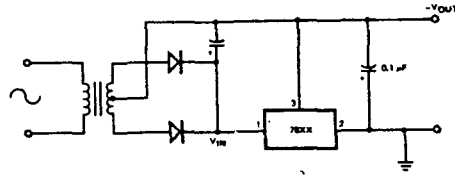


± TRACKING VOLTAGE REGULATOR

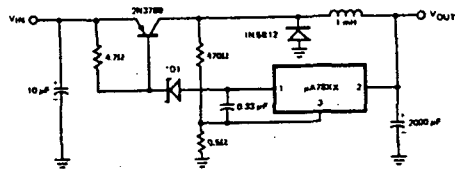
TYPICAL APPLICATIONS (Cont'd)



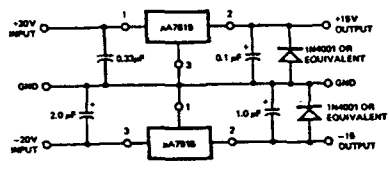
POSITIVE AND NEGATIVE REGULATOR



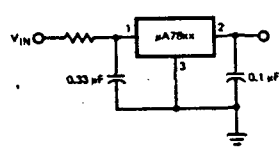
NEGATIVE OUTPUT VOLTAGE CIRCUIT



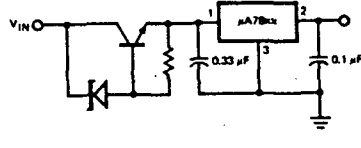
SWITCHING REGULATOR



DUAL SUPPLY  
OPERATIONAL AMPLIFIER SUPPLY ( $\pm 15$  V @ 1.0 A)



HIGH INPUT VOLTAGE CIRCUITS



# $\mu$ A109 • $\mu$ A209

## 5 VOLT REGULATOR FAIRCHILD LINEAR INTEGRATED CIRCUITS

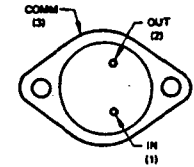
**GENERAL DESCRIPTION** — The 109 and 209 are complete Five Volt Regulators constructed using the Fairchild Planar\* epitaxial process. These regulators employ internal current limiting, thermal shutdown and safe-area compensation making them essentially indestructible. They are intended for use as local regulators, eliminating noise and distribution problems associated with single point regulation. If adequate heat sinking is provided, they can provide over 1A output current. The 109 and 209 are intended primarily for use with TTL and DTL logic and are completely specified under worst case conditions to match the power supply requirements of these logic families. In addition to use as a fixed 5 V regulator, these devices can be used with external components to obtain adjustable output voltages and currents and as the power pass element in precision regulators.

- OUTPUT CURRENT IN EXCESS OF 1 A
- SPECIFIED TO MATCH WORST CASE TTL AND DTL REQUIREMENTS
- NO EXTERNAL COMPONENTS
- INTERNAL THERMAL OVERLOAD PROTECTION
- OUTPUT TRANSISTOR SAFE-AREA COMPENSATION

**ABSOLUTE MAXIMUM RATINGS**

Input Voltage	35 V
Internal Power Dissipation	Internally Limited
Storage Temperature Range	-65°C to +150°C
Operating Junction Temperature Range	
Military Grade ( $\mu$ A109)	-55°C to +150°C
Industrial Grade ( $\mu$ A209)	-25°C to +150°C
Lead Temperature (Soldering, 60 seconds)	300°C

**CONNECTION DIAGRAM  
TO-3 PACKAGE  
(TOP VIEW)**

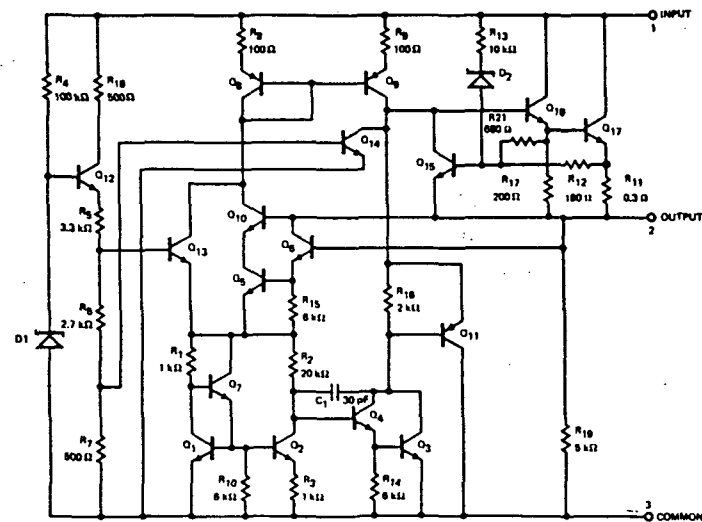


Case is connected to ground.

**ORDER INFORMATION**

TYPE	PART NO.
$\mu$ A109	$\mu$ A109KM
$\mu$ A209	$\mu$ A209KM

**EQUIVALENT CIRCUIT**



\*Planar is a patented Fairchild process.

# μA7900 SERIES

## 3-TERMINAL NEGATIVE VOLTAGE REGULATORS

### FAIRCHILD LINEAR INTEGRATED CIRCUITS

**GENERAL DESCRIPTION** - The μA7900 series of monolithic 3-Terminal Negative Regulators is manufactured using the Fairchild Planar\* epitaxial process. These negative regulators are intended as complements to the popular μA7800 series of positive voltage regulators, and they are available in the same voltage options from -5 to -24 V. The 7900s employ internal current limiting, safe-area protection, and thermal shutdown, making them virtually indestructible.

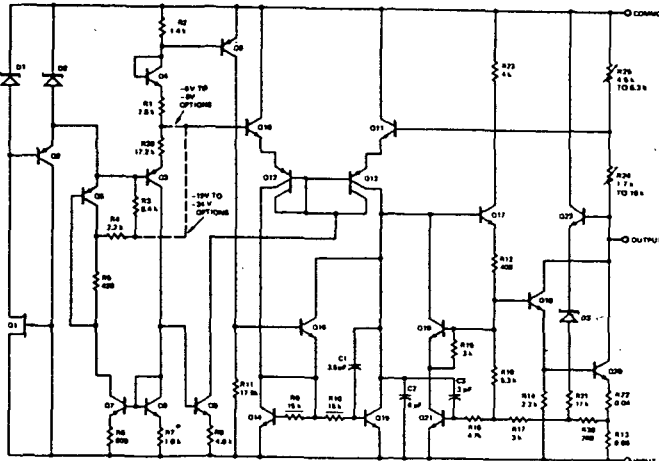
- OUTPUT CURRENT IN EXCESS OF 1 A
- INTERNAL THERMAL OVERLOAD PROTECTION
- INTERNAL SHORT CIRCUIT CURRENT LIMITING
- OUTPUT TRANSISTOR SAFE AREA COMPENSATION
- AVAILABLE IN THE TO-220 AND THE TO-3 PACKAGE
- OUTPUT VOLTAGES ARE 5, 6, 8, 12, 15, 18 AND 24 V

#### ABSOLUTE MAXIMUM RATINGS

Input Voltage (5 V through 18 V)	-35 V
(24 V)	-40 V
Internal Power Dissipation	Internally Limited
Storage Temperature Range	
TO-3 (Al. or Steel)	-65°C to +150°C
TO-220	-55°C to +150°C
Operating Junction Temperature Range	
Military (μA7900)	-55°C to +150°C
Commercial (μA7900C)	0°C to +150°C
Lead Temperature	
TO-3 (Soldering, 60 s)	300°C
TO-220 (Soldering, 10 s)	230°C

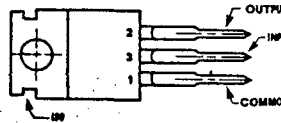
NOTE: The convention for Negative Regulators is the Algebraic value, thus -15 is less than -10 V.

#### EQUIVALENT CIRCUIT



\*Planar is a patented Fairchild process.

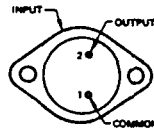
#### CONNECTION DIAGRAMS TO-220 PACKAGE (TOP VIEW)



#### ORDER INFORMATION

OUTPUT VOLTAGE	TYPE	PART NO.
-5 V	μA7905C	μA7905UC
-6 V	μA7906C	μA7906UC
-8 V	μA7908C	μA7908UC
-12 V	μA7912C	μA7912UC
-15 V	μA7915C	μA7915UC
-18 V	μA7918C	μA7918UC
-24 V	μA7924C	μA7924UC

#### TO-3 PACKAGE (TOP VIEW)



#### ORDER INFORMATION

OUTPUT VOLTAGE	TYPE	PART NO.
-5 V	μA7906	μA7906KM
-6 V	μA7906	μA7906KM
-8 V	μA7908	μA7908KM
-12 V	μA7912	μA7912KM
-15 V	μA7915	μA7915KM
-18 V	μA7918	μA7918KM
-24 V	μA7924	μA7924KM
-5 V	μA7906C	μA7906KC
-6 V	μA7906C	μA7906KC
-8 V	μA7908C	μA7908KC
-12 V	μA7912C	μA7912KC
-15 V	μA7915C	μA7915KC
-18 V	μA7918C	μA7918KC
-24 V	μA7924C	μA7924KC

## FAIRCHILD • μA7900 SERIES

### μA7908

**ELECTRICAL CHARACTERISTICS:**  $V_{IN} = -14 V$ ,  $I_{OUT} = 500 mA$ ,  $C_{IN} = 2 \mu F$ ,  $C_{OUT} = 1 \mu F$ ,  $-55^\circ C < T_J < 150^\circ C$ , unless otherwise specified.

CHARACTERISTICS	CONDITIONS (Note 1)	MIN	TYP	MAX	UNITS
Output Voltage	$T_J = 25^\circ C$	-7.7	-8.0	-8.3	V
Line Regulation	$T_J = 25^\circ C$	$-10.5 V < V_{IN} < -25 V$	6.0	80	mV
		$-11 V < V_{IN} < -17 V$	2.0	40	mV
Load Regulation	$T_J = 25^\circ C$	$5 mA < I_{OUT} < 1.5 A$	12	80	mV
		$250 mA < I_{OUT} < 750 mA$	4.0	40	mV
Output Voltage	$-11.5 V < V_{IN} < -23 V$ $5 mA < I_{OUT} < 1.0 A$ $p < 15 W$	-7.6		-8.4	V
Quiescent Current	$T_J = 25^\circ C$	1.0	2.0		mA
Quiescent Current Change	with line with load	$-11.5 V < V_{IN} < -25 V$		1.0	mA
		$5 mA < I_{OUT} < 1.0 A$		0.5	mA
Output Noise Voltage	$T_A = 25^\circ C$ , 10 Hz $< f < 100$ kHz		25	80	$\mu V/V_{OUT}$
Ripple Rejection	$f = 120$ Hz, $-11.5 V < V_{IN} < -21.5 V$	54	60		dB
Dropout Voltage	$I_{OUT} = 1.0 A$ , $T_J = 25^\circ C$	1.1	2.3		V
Peak Output Current	$T_J = 25^\circ C$	1.3	2.1	3.3	A
Average Temperature Coefficient of Output Voltage	$I_{OUT} = 5 mA$ , $-55^\circ C < T_J < 150^\circ C$			0.3	$mV/^\circ C/V_{OUT}$
Short Circuit Current	$V_{IN} = -35 V$ , $T_J = 25^\circ C$			1.2	A

### μA7908C

**ELECTRICAL CHARACTERISTICS:**  $V_{IN} = -14 V$ ,  $I_{OUT} = 500 mA$ ,  $C_{IN} = 2 \mu F$ ,  $C_{OUT} = 1 \mu F$ ,  $0^\circ C < T_J < 125^\circ C$ , unless otherwise specified.

CHARACTERISTICS	CONDITIONS (Note 1)	MIN	TYP	MAX	UNITS	
Output Voltage	$T_J = 25^\circ C$	-7.7	-8.0	-8.3	V	
Line Regulation	$T_J = 25^\circ C$	$-10.5 V < V_{IN} < -25 V$	6.0	160	mV	
		$-11 V < V_{IN} < -17 V$	2.0	80	mV	
Load Regulation	$T_J = 25^\circ C$	$5 mA < I_{OUT} < 1.5 A$		12	160	mV
		$250 mA < I_{OUT} < 750 mA$		4.0	80	mV
Output Voltage	$-10.5 V < V_{IN} < -23 V$ $5 mA < I_{OUT} < 1.0 A$ $p < 15 W$	-7.6		-8.4	V	
Quiescent Current	$T_J = 25^\circ C$	1.0	2.0		mA	
Quiescent Current Change	with line with load	$-10.5 V < V_{IN} < -25 V$		1.0	mA	
		$5 mA < I_{OUT} < 1.0 A$		0.5	mA	
Output Noise Voltage	$T_A = 25^\circ C$ , 10 Hz, $< f < 100$ kHz		200		$\mu V$	
Ripple Rejection	$f = 120$ Hz, $-11.5 V < V_{IN} < -21.5 V$	54	60		dB	
Dropout Voltage	$I_{OUT} = 1.0 A$ , $T_J = 25^\circ C$		1.1		V	
Peak Output Current	$T_J = 25^\circ C$		2.1		A	
Average Temperature Coefficient of Output Voltage	$I_{OUT} = 5 mA$ , $0^\circ C < T_J < 125^\circ C$			-0.6	$mV/^\circ C$	

NOTE: All characteristics except noise voltage and ripple rejection ratio are measured using pulse techniques ( $t_w < 10$  ms, duty cycle  $< 5\%$ ). Output voltage changes due to changes in internal temperature must be taken into account separately.

ORIGINAL



**FAIRCHILD •  $\mu$ A7900 SERIES**

**$\mu$ A7912**

**ELECTRICAL CHARACTERISTICS:**  $V_{IN} = -19$  V,  $I_{OUT} = 500$  mA,  $C_{IN} = 2 \mu$ F,  $C_{OUT} = 1 \mu$ F,  $-55^\circ\text{C} < T_J < 150^\circ\text{C}$ , unless otherwise specified.

CHARACTERISTICS	CONDITIONS (Note 1)	MIN	TYP	MAX	UNITS
Output Voltage	$T_J = 25^\circ\text{C}$	-11.5	-12.0	-12.5	V
Line Regulation	$T_J = 25^\circ\text{C}$ $-14.5 \text{ V} < V_{IN} < -30 \text{ V}$ $-16 \text{ V} < V_{IN} < -22 \text{ V}$		10	120	mV
			3.0	60	mV
Load Regulation	$T_J = 25^\circ\text{C}$ $5 \text{ mA} < I_{OUT} < 1.5 \text{ A}$ $250 \text{ mA} < I_{OUT} < 750 \text{ mA}$		12	120	mV
			4.0	60	mV
Output Voltage	$-15.5 \text{ V} < V_{IN} < -27 \text{ V}$ $5 \text{ mA} < I_{OUT} < 1.0 \text{ A}$ $p < 15 \text{ W}$	-11.4		-12.6	V
Quiescent Current	$T_J = 25^\circ\text{C}$		1.5	3.0	mA
Quiescent Current Change	with line $-15 \text{ V} < V_{IN} < -30 \text{ V}$			1.0	mA
	with load $5 \text{ mA} < I_{OUT} < 1.0 \text{ A}$			0.5	mA
Output Noise Voltage	$T_A = 25^\circ\text{C}$ , $10 \text{ Hz} < f < 100 \text{ kHz}$		25	80	$\mu\text{V}/V_{OUT}$
Ripple Rejection	$f = 120 \text{ Hz}$ , $-15 \text{ V} < V_{IN} < -25 \text{ V}$	54	60		dB
Dropout Voltage	$I_{OUT} = 1.0 \text{ A}$ , $T_J = 25^\circ\text{C}$		1.1	2.3	V
Peak Output Current	$T_J = 25^\circ\text{C}$	1.3	2.1	3.3	A
Average Temperature Coefficient of Output Voltage	$I_{OUT} = 5 \text{ mA}$ , $-55^\circ\text{C} < T_J < 150^\circ\text{C}$			0.3	$\text{mV}/^\circ\text{C}$ $V_{OUT}$
Short Circuit Current	$V_{IN} = -35 \text{ V}$ , $T_J = 25^\circ\text{C}$			1.2	A

**$\mu$ A7912C**

**ELECTRICAL CHARACTERISTICS:**  $V_{IN} = -19$  V,  $I_{OUT} = 500$  mA,  $C_{IN} = 2 \mu$ F,  $C_{OUT} = 1 \mu$ F,  $0^\circ\text{C} < T_J < 125^\circ\text{C}$ , unless otherwise specified.

CHARACTERISTICS	CONDITIONS (Note 1)	MIN	TYP	MAX	UNITS
Output Voltage	$T_J = 25^\circ\text{C}$	-11.5	-12.0	-12.5	V
Line Regulation	$T_J = 25^\circ\text{C}$ $-14.5 \text{ V} < V_{IN} < -30 \text{ V}$ $-16 \text{ V} < V_{IN} < -22 \text{ V}$		10	240	mV
			3.0	120	mV
Load Regulation	$T_J = 25^\circ\text{C}$ $5 \text{ mA} < I_{OUT} < 1.5 \text{ A}$ $250 \text{ mA} < I_{OUT} < 750 \text{ mA}$		12	240	mV
			4.0	120	mV
Output Voltage	$-14.5 \text{ V} < V_{IN} < -27 \text{ V}$ $5 \text{ mA} < I_{OUT} < 1.0 \text{ A}$ $p < 15 \text{ W}$	-11.4		-12.6	V
Quiescent Current	$T_J = 25^\circ\text{C}$		1.5	3.0	mA
Quiescent Current Change	with line $-14.5 \text{ V} < V_{IN} < -30 \text{ V}$			1.0	mA
	with load $5 \text{ mA} < I_{OUT} < 1.0 \text{ A}$			0.5	mA
Output Noise Voltage	$T_A = 25^\circ\text{C}$ , $10 \text{ Hz} < f < 100 \text{ kHz}$		300		$\mu\text{V}$
Ripple Rejection	$f = 120 \text{ Hz}$ , $-15 \text{ V} < V_{IN} < -25 \text{ V}$	54	60		dB
Dropout Voltage	$I_{OUT} = 1.0 \text{ A}$ , $T_J = 25^\circ\text{C}$		1.1		V
Peak Output Current	$T_J = 25^\circ\text{C}$		2.1		A
Average Temperature Coefficient of Output Voltage	$I_{OUT} = 5 \text{ mA}$ , $0^\circ\text{C} < T_J < 125^\circ\text{C}$			-0.8	$\text{mV}/^\circ\text{C}$

**NOTE:**

1. All characteristics except noise voltage and ripple rejection ratio are measured using pulse techniques ( $t_w < 10$  ms, duty cycle  $< 5\%$ ). Output voltage changes due to changes in internal temperature must be taken into account separately.

**FAIRCHILD •  $\mu$ A7900 SERIES**

**$\mu$ A7905**

**ELECTRICAL CHARACTERISTICS:**  $V_{IN} = -10$  V,  $I_{OUT} = 500$  mA,  $C_{IN} = 2 \mu$ F,  $C_{OUT} = 1 \mu$ F,  $-55^\circ\text{C} < T_J < 150^\circ\text{C}$ , unless otherwise specified.

CHARACTERISTICS	CONDITIONS (Note 1)	MIN	TYP	MAX	UNITS
Output Voltage	$T_J = 25^\circ\text{C}$	-4.8	-5.0	-5.2	V
Line Regulation	$T_J = 25^\circ\text{C}$ $-7 \text{ V} < V_{IN} < -25 \text{ V}$ $-8 \text{ V} < V_{IN} < -12 \text{ V}$		3	50	mV
			1	25	mV
Load Regulation	$T_J = 25^\circ\text{C}$ $5 \text{ mA} < I_{OUT} < 1.5 \text{ A}$ $250 \text{ mA} < I_{OUT} < 750 \text{ mA}$		15	50	mV
			5	25	mV
Output Voltage	$-8.0 \text{ V} < V_{IN} < -20 \text{ V}$ $5 \text{ mA} < I_{OUT} < 1.0 \text{ A}$ $p < 15 \text{ W}$	-4.70		-5.30	V
Quiescent Current	$T_J = 25^\circ\text{C}$		1.0	2.0	mA
Quiescent Current Change	with line $-8 \text{ V} < V_{IN} < -25 \text{ V}$			1.3	mA
	with load $5 \text{ mA} < I_{OUT} < 1.0 \text{ A}$			0.5	mA
Output Noise Voltage	$T_A = 25^\circ\text{C}$ , $10 \text{ Hz} < f < 100 \text{ kHz}$		25	80	$\mu\text{V}/V_{OUT}$
Ripple Rejection	$f = 120 \text{ Hz}$ , $-8 \text{ V} < V_{IN} < -18 \text{ V}$	54	60		dB
Dropout Voltage	$I_{OUT} = 1.0 \text{ A}$ , $T_J = 25^\circ\text{C}$		1.1	2.3	V
Peak Output Current	$T_J = 25^\circ\text{C}$	1.3	2.1	3.3	A
Average Temperature Coefficient of Output Voltage	$I_{OUT} = 5 \text{ mA}$ , $-55^\circ\text{C} < T_J < 150^\circ\text{C}$			0.3	$\text{mV}/^\circ\text{C}$ $V_{OUT}$
Short Circuit Current	$V_{IN} = -35 \text{ V}$ , $T_J = 25^\circ\text{C}$			1.2	A

**$\mu$ A7905C**

**ELECTRICAL CHARACTERISTICS:**  $V_{IN} = -10$  V,  $I_{OUT} = 500$  mA,  $C_{IN} = 2 \mu$ F,  $C_{OUT} = 1 \mu$ F,  $0^\circ\text{C} < T_J < 125^\circ\text{C}$ , unless otherwise specified.

CHARACTERISTICS	CONDITIONS (Note 1)	MIN	TYP	MAX	UNITS
Output Voltage	$T_J = 25^\circ\text{C}$	-4.8	-5.0	-5.2	V
Line Regulation	$T_J = 25^\circ\text{C}$ $-7 \text{ V} < V_{IN} < -25 \text{ V}$ $-8 \text{ V} < V_{IN} < -12 \text{ V}$		3.0	100	mV
			1.0	50	mV
Load Regulation	$T_J = 25^\circ\text{C}$ $5 \text{ mA} < I_{OUT} < 1.5 \text{ A}$ $250 \text{ mA} < I_{OUT} < 750 \text{ mA}$		15	100	mV
			5.0	50	mV
Output Voltage	$-7 \text{ V} < V_{IN} < -20 \text{ V}$ $5 \text{ mA} < I_{OUT} < 1.0 \text{ A}$ $p < 15 \text{ W}$	-4.75		-5.25	V
Quiescent Current	$T_J = 25^\circ\text{C}$		1.0	2.0	mA
Quiescent Current Change	with line $-7 \text{ V} < V_{IN} < -25 \text{ V}$			1.3	mA
	with load $5 \text{ mA} < I_{OUT} < 1.0 \text{ A}$			0.5	mA
Output Noise Voltage	$T_A = 25^\circ\text{C}$ , $10 \text{ Hz} < f < 100 \text{ kHz}$		125		$\mu\text{V}$
Ripple Rejection	$f = 120 \text{ Hz}$ , $-8 \text{ V} < V_{IN} < -18 \text{ V}$	54	60		dB
Dropout Voltage	$I_{OUT} = 1.0 \text{ A}$ , $T_J = 25^\circ\text{C}$		1.1		V
Peak Output Current	$T_J = 25^\circ\text{C}$		2.1		A
Average Temperature Coefficient of Output Voltage	$I_{OUT} = 5 \text{ mA}$ , $0^\circ\text{C} < T_J < 125^\circ\text{C}$			-0.4	$\text{mV}/^\circ\text{C}$

**NOTE:**

1. All characteristics except noise voltage and ripple rejection ratio are measured using pulse techniques ( $t_w < 10$  ms, duty cycle  $< 5\%$ ). Output voltage changes due to changes in internal temperature must be taken into account separately.

DESIGN CONSIDERATIONS

The  $\mu$ A7900 fixed voltage regulator series has thermal overload protection from excessive power, internal short circuit protection which limits the circuit's maximum current, and output transistor safe area compensation for reducing the output current as the voltage across the transistor is increased.

Although the internal power dissipation is limited, the junction temperature must be kept below the maximum specified temperature (125°C for 7900, 125°C for 7900C) in order to meet data sheet specifications. To calculate the maximum junction temperature or heat sink required the following thermal resistance values should be used:

Package	TYP $\theta_{JC}$	MAX $\theta_{JC}$	TYP $\theta_{JA}$	MAX $\theta_{JA}$
TO-3	3.5°C/W	5.5°C/W	40°C/W	45°C/W
TO-220	3.0°C/W	5.0°C/W	60°C/W	65°C/W

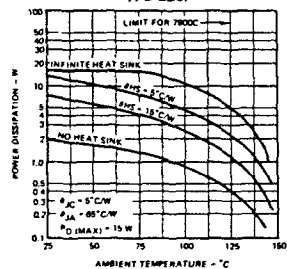
$$P_D(\text{MAX}) = \frac{T_J(\text{MAX}) - T_A}{\theta_{JC} + \theta_{CA}} \text{ or } \frac{T_J(\text{MAX}) - T_A}{\theta_{JA}} \text{ (Without a heat sink)}$$

$$\theta_{CA} = \theta_{CS} + \theta_{SA}$$

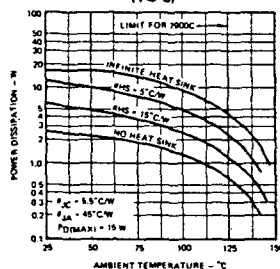
$$\text{Solving for } T_J: T_J = T_A + P_D(\theta_{JC} + \theta_{CA}) \text{ or } T_A + P_D\theta_{JA} \text{ (Without heat sink)}$$

- Where  $T_J$  = Junction Temperature  $\theta_{JC}$  = Junction to Case Thermal Resistance  
 $T_A$  = Ambient Temperature  $\theta_{CA}$  = Case to Ambient Thermal Resistance  
 $P_D$  = Power Dissipation  $\theta_{CS}$  = Case to Heat Sink Thermal Resistance  
 $\theta_{JA}$  = Junction to Ambient Thermal Resistance  $\theta_{SA}$  = Heat Sink to Ambient Thermal Resistance

WORST CASE POWER DISSIPATION AS A FUNCTION OF AMBIENT TEMPERATURE (TO-220)

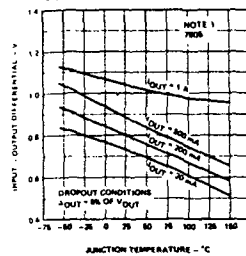


WORST CASE POWER DISSIPATION AS A FUNCTION OF AMBIENT TEMPERATURE (TO-3)

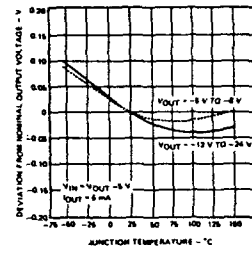


TYPICAL PERFORMANCE CURVES

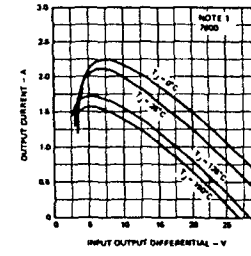
DROPOUT VOLTAGE AS A FUNCTION OF JUNCTION TEMPERATURE



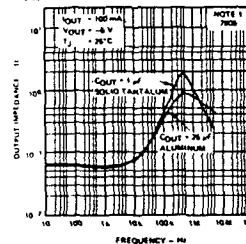
OUTPUT VOLTAGE AS A FUNCTION OF JUNCTION TEMPERATURE



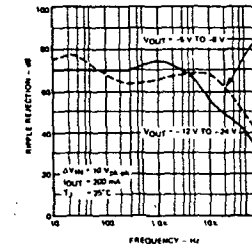
PEAK OUTPUT CURRENT AS A FUNCTION OF INPUT-OUTPUT DIFFERENTIAL VOLTAGE



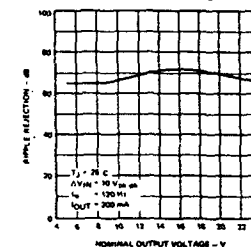
OUTPUT IMPEDANCE AS A FUNCTION OF FREQUENCY



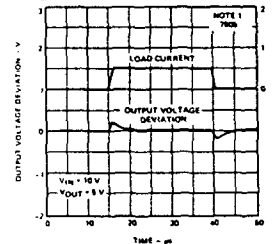
RIPPLE REJECTION AS A FUNCTION OF FREQUENCY



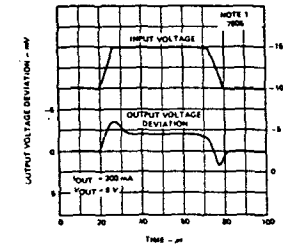
RIPPLE REJECTION AS A FUNCTION OF OUTPUT VOLTAGE



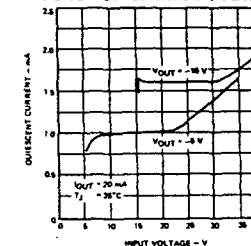
LOAD TRANSIENT RESPONSE



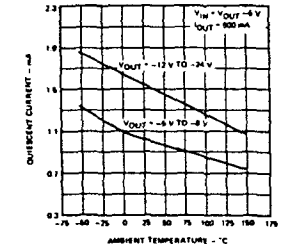
LINE TRANSIENT RESPONSE



QUIESCENT CURRENT AS A FUNCTION OF INPUT VOLTAGE



QUIESCENT CURRENT AS A FUNCTION OF TEMPERATURE

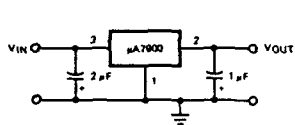


NOTE 1: The other  $\mu$ A7900 series devices have similar performance curves.

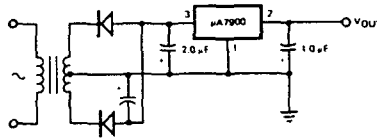
TYPICAL APPLICATIONS

Bypass capacitors are recommended for stable operation of the  $\mu$ A7900 series of regulators over the input voltage and output current range. Output bypass capacitors will improve the transient response of the regulator.

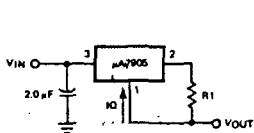
The bypass capacitors, (2  $\mu$ F on the input, 1  $\mu$ F on the output) should be ceramic or solid tantalum which have good high frequency characteristics. If aluminum electrolytics are used, their values should be 10  $\mu$ F or larger. The bypass capacitors should be mounted with the shortest leads, and if possible, directly across the regulator terminals.



FIXED OUTPUT REGULATOR

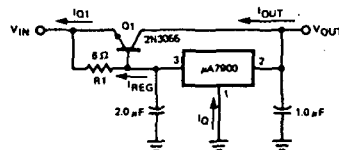


NEGATIVE OUTPUT VOLTAGE CIRCUIT



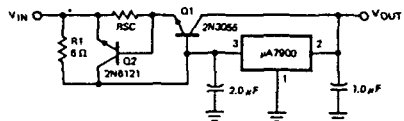
$$\text{OUTPUT CURRENT} = \frac{5.0 \text{ V}}{R1} + I_Q$$

BASIC CURRENT REGULATOR



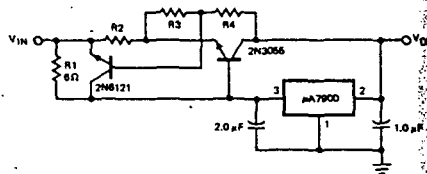
$$R1 = \frac{V_{BE}(Q1)}{I_{REG}} \quad I_{Q1} = \beta(Q1) I_{REG}$$

HIGH CURRENT VOLTAGE REGULATOR

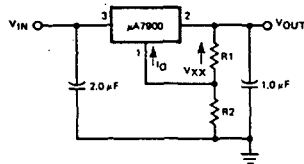


$$R_{SC} = \frac{V_{BE}(Q2)}{I_{SC}}$$

HIGH OUTPUT CURRENT, SHORT CIRCUIT PROTECTED

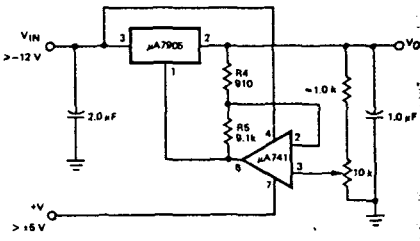


HIGH OUTPUT CURRENT, FOLDBACK CURRENT LIMITED



$$|V_{OUT}| = V_{XX} \left(1 + \frac{R2}{R1}\right) + I_Q R2$$

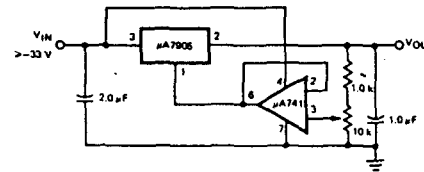
VARIABLE OUTPUT VOLTAGE REGULATOR



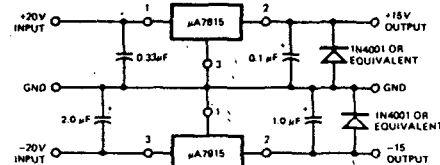
VARIABLE OUTPUT VOLTAGE, -0.5 V TO -10 V

TYPICAL APPLICATIONS (Cont'd)

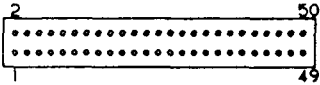
VARIABLE OUTPUT VOLTAGE, -30 V TO -7 V



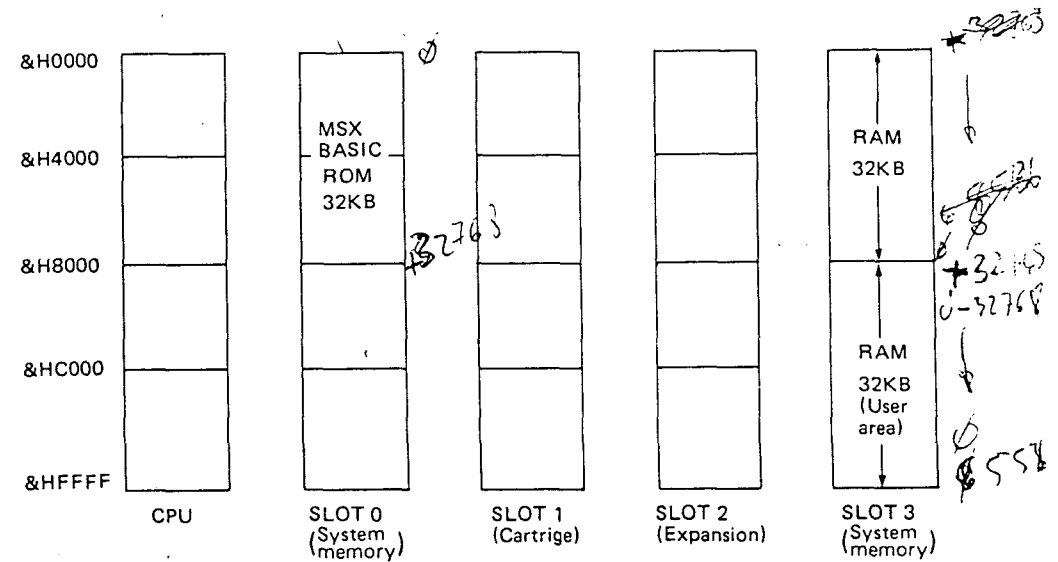
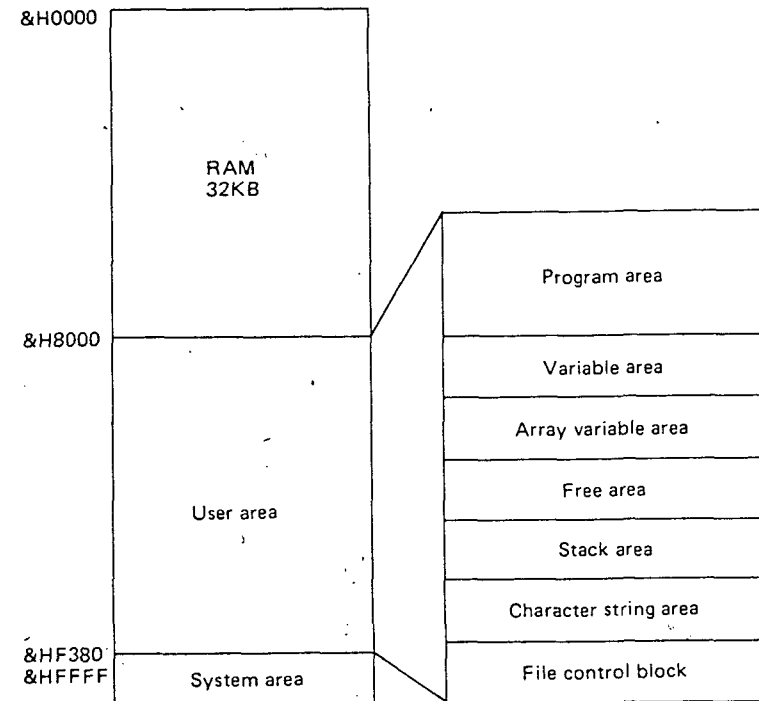
OPERATIONAL AMPLIFIER SUPPLY ( $\pm 15 \text{ V @ } 1.0 \text{ A}$ )



• ROM Cartridge Slot/Expansion Bus

Pin No.	Name	Pin Connection
1	CST	(Expansion Bus) 
2	CS2	
3	CS12	
4	SLTSL	
6	RF SH	
7	WAIT	
8	INT	
9	MI	
10	BUSDIR	
11	IORQ	
12	MERQ	
13	WR	
14	RD	
15	RESET	
17~32	A0~A15	
33~40	D0~D7	
41	GND	
42	CLOCK	
43	GND	
44, 46	SW1, SW2	
45, 47	+5V	
48	+12V	
49	SUNDIN	
50	-12V	

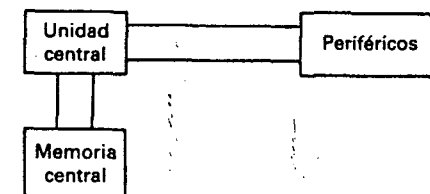
12-3 MEMORY MAP



# 1. La programación en ensamblador

## 1.1. Introducción: estructura interna del microordenador

Un sistema microordenador consta de tres elementos fundamentales que son: la unidad central, la memoria central y los periféricos. Estos elementos están entrelazados entre ellos por buses, los cuales están constituidos por cierto número de conexiones eléctricas destinadas a permitir la transferencia de informaciones.



La unidad central es el núcleo del microordenador; ella es quien controla la transferencia de datos con la memoria central y con los periféricos.

El componente fundamental de la unidad central es el microprocesador. Es el que realiza la mayor parte de las funciones de la unidad central, o sea los cálculos. En el SPECTRUM, este microprocesador es fabricado por ZILOG y se llama Z 80.

La memoria central es un elemento esencial del microordenador. Sin ella éste no podría funcionar. Está destinada a contener los programas ejecutados y los datos manipulados por la unidad central.

### 1.1.1. La memoria central

La memoria central está dividida en cierta cantidad de posiciones pudiendo contener un dato numérico entero comprendido entre 0 y 255. Cada una de estas posiciones tiene 8 bits y se llama *octeto* o *byte* en inglés. El bit es el elemento más pequeño de la memoria. Solamen-

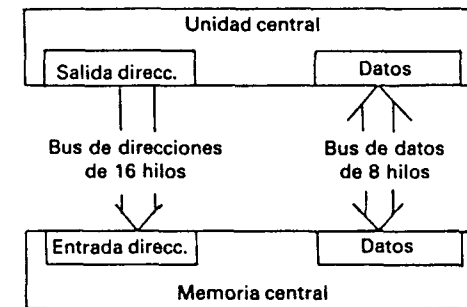
te puede tomar dos valores: 0 (estado bajo: 0 V) y 1 (estado alto: 5 V), lo cual permite memorizar un número entero comprendido entre 0 y 1. Agrupando 8 bits se obtiene un octeto que puede memorizar un número comprendido entre 0000 0000 y 1111 1111 en binario,\* o sea 0 y 255 en decimal. Se ha tomado el hábito de utilizar la base 16 (representación hexadecimal) para indicar el valor de un octeto. Con esta base el valor del octeto varía desde 00 a FF. Esta representación corresponde lo mejor posible al reparto en octetos, ya que permite codificar un octeto sólo mediante dos símbolos, y cualquier número hexadecimal de dos cifras puede ser colocado en un octeto.

De manera simbólica podemos decir que la memoria está constituida por una cantidad de octetos colocados uno detrás de otro formando cadena. A cada octeto se le asocia un número de orden que representa su posición dentro de la cadena. A este número se le llama *dirección* del octeto. El primero tendrá la dirección 0, el segundo la dirección 1 y así sucesivamente hasta el último. Esta dirección servirá para seleccionar cualquiera de los octetos de la memoria central.

El microprocesador Z 80 es capaz de manipular direcciones de memoria comprendidas entre 0 y 65535 ( $2^{16} - 1$ ). Para ello dispone de un conjunto de 16 salidas de dirección. Estas salidas sólo pueden tener los valores 0 y 1; gracias a ello se pueden formar todas las combinaciones de números binarios comprendidos entre 0000 0000 0000 0000 y 1111 1111 1111 1111, o sea 0 y 65535 en decimal, o 0000 y FFFF en hexadecimal. Estas 16 salidas están ligadas a 16 entradas de dirección de la memoria central a través del bus de direcciones. Aplicando una combinación binaria a estas salidas de dirección, el microprocesador puede seleccionar uno de los octetos de la memoria central. Para utilizar este octeto el microprocesador tiene 8 bornes de datos conectados a 8 bornes de datos de la memoria central mediante el bus de datos. Así, el octeto de la memoria central seleccionado por el microprocesador podrá transitar entre la memoria central y el microprocesador. Los bornes de datos y el bus de datos son bidireccionales para permitir la transferencia de octetos entre el microprocesador y la memoria (escritura en memoria), o bien entre la memoria y el microprocesador (lectura en memoria).

Existen dos tipos de memoria en el SPECTRUM. El primero, la memoria muerta (en inglés ROM = Read Only Memory), que solamente funciona en lectura. Su contenido es memorizado una sola vez por todas; no puede ser modificado por el microprocesador. Su valor es de 16 384 octetos. Para indicar este valor generalmente se utiliza el kiloocteto, que es igual a  $2^{10}$  octetos, o sea 1 024. El valor de la memoria muerta (ROM) es pues de 16 K (abreviación de kiloocteto).

\*En el anexo 1 se da una explicación sobre las bases de numeración.



El segundo tipo de memoria es la viva (en inglés RAM = Random Access Memory), que trabaja tanto en lectura como en escritura. Los datos registrados en esta memoria sólo son memorizados cuando el microordenador está alimentado por corriente eléctrica. En el SPECTRUM, el valor de esta memoria es de 16 K o 48 K, según la configuración del modelo.

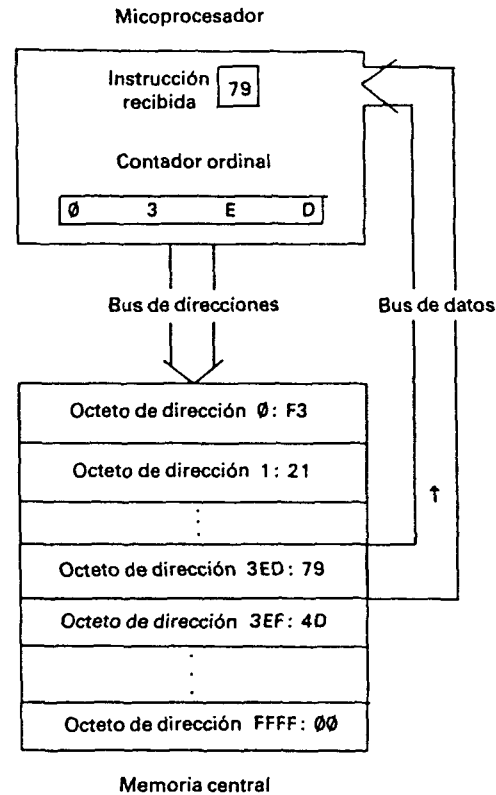
### 1.1.2. El microprocesador

El microprocesador Z 80 es una máquina capaz de ejecutar cierta cantidad de *instrucciones*. Estas se distinguen mediante un código, el cual se introduce en 1, 2, 3 o 4 octetos. Las operaciones ordenadas por estas instrucciones generalmente son muy simples (adición, sustracción, operación lógica, transferencia de memoria).

El microprocesador ha sido concebido para ejecutar secuencialmente una serie de instrucciones almacenadas en la memoria central, ejecutando así un programa almacenado en la memoria. Para ello dispone de una memoria interna de 16 bits llamada contador ordinal. Este contador ordinal sirve para memorizar la dirección de la próxima instrucción a ejecutar. El ciclo completo para ejecutar una instrucción es el siguiente.

1. El contador ordinal se encuentra ligado a las salidas de dirección para seleccionar el octeto que contiene el código de la instrucción a ejecutar.
2. El octeto seleccionado llega al microprocesador por el bus de datos.
3. El microprocesador descodifica la instrucción y ejecuta la acción correspondiente.
4. El contenido del contador ordinal queda aumentado en el número de octetos de la instrucción ejecutada, de manera que contenga la dirección de la instrucción siguiente.

5. El proceso se repite a partir de la etapa 1 para la instrucción siguiente.



La instrucción situada en la dirección 3ED (valor del contador ordinal) es transferida hacia el Z 80 que la ejecutará. Aumentará en uno el contador ordinal y ejecutará la instrucción siguiente situada en la dirección 3EE (hexadecimal).

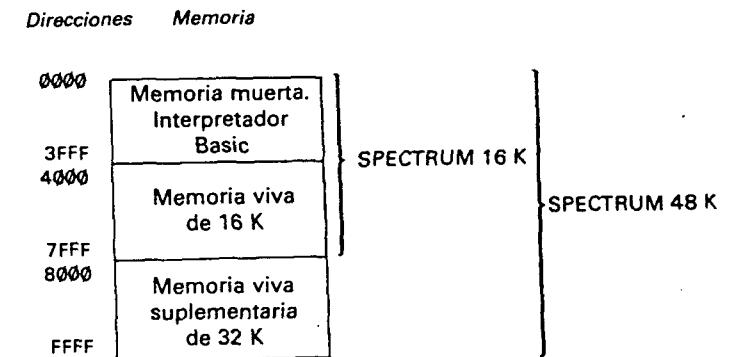
De esta manera, el microprocesador ejecuta cada instrucción del programa guardado en la memoria central con la forma de códigos binarios. Así pues, puede decirse que es un programa escrito en *lenguaje máquina* (o código máquina) que es el único lenguaje capaz de interpretar el microprocesador.

El lenguaje Basic es totalmente incomprendible para el microprocesador que no sabe ejecutar una orden como PRINT. Para la utilización del lenguaje Basic en el microordenador ha sido necesario crear una interfase entre el lenguaje binario manipulado por el microprocesador y las órdenes del lenguaje Basic. Este lenguaje, escrito naturalmente en lenguaje máquina, y que es el único que puede comprender

el microprocesador, está colocado en los 16 K de memoria muerta. De esta forma está constantemente en las memorias del ordenador. Cuando se conecta a tensión se activa automáticamente por la puesta a cero del contador ordinal. El interpretador, que empieza en la dirección cero, es así puesto en marcha al conectar la tensión. El usuario tiene entonces la impresión de trabajar sobre una máquina que solamente comprende el lenguaje Basic.

Todo programa escrito en lenguaje máquina (como el interpretador Basic) tiene necesidad de manipular cierto número de datos. Debido a ello se utiliza la memoria viva para escribir, leer o modificar las variables que utiliza. Estas variables están constituidas por octetos de la memoria central, y pueden transitar entre el microprocesador y la memoria. El microprocesador, que no contiene más que un bus de datos de 8 hilos, no puede leer más de un octeto a la vez. Así, se dice que es un *microprocesador de 8 bits*. Para manipular variables que contengan varios octetos será necesario descomponerlas. Descubrimos así el interés de los microprocesadores de 16 bits y de 32 bits, que permiten manipular de una sola vez variables de tamaño más grande. Esto comporta más rapidez y facilidades en el desarrollo de los programas.

Todo programa en lenguaje máquina, que no sea el interpretador del Basic, deberá colocarse en la memoria viva. Demos por este motivo la distribución de la ocupación de las memorias (en inglés: memory map) en el ZX SPECTRUM.



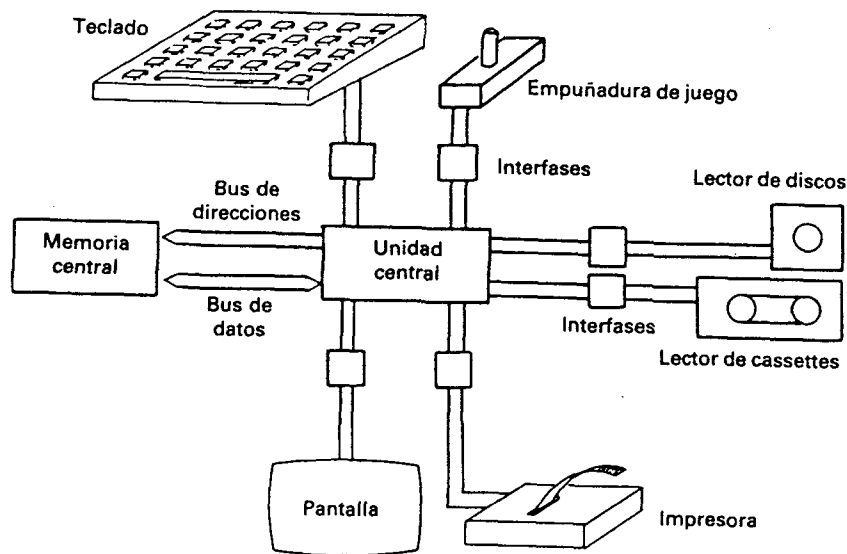
### 1.1.3. Los periféricos

Los periféricos son los órganos de enlace entre el ordenador y el mundo exterior. No son primordialmente necesarios para el funcionamiento del ordenador, pero sin ellos no sirve para nada, pues por sí solo no puede comunicar sus resultados al usuario.

Existen principalmente tres tipos de periféricos:

- Los periféricos que funcionan como entradas y que recogen las órdenes del usuario. Como es el caso del teclado y los mandos de juego.
- Los periféricos que funcionan como salidas y sirven para visualizar los resultados obtenidos por el ordenador. Como son la pantalla de video y la impresora.
- Los periféricos que funcionan como entradas y salidas; son el lector de cassettes y las unidades de disco. El intercambio de informaciones entre el periférico y el ordenador se realiza así en forma bidireccional.

Para permitir la utilización de periféricos, el microordenador contiene interfases materiales que aseguran la conexión entre la unidad central, el bus de datos, el bus de direcciones y los periféricos. Algunos de los subprogramas escritos en lenguaje máquina introducidos en la ROM permiten la gestión software de los periféricos. Dichos programas son calificados de «rutinas de gestión de las entradas/salidas».



## 1.2. Ventajas e inconvenientes del lenguaje máquina

El lenguaje máquina presenta dos ventajas importantes con respecto a un lenguaje de alto nivel como es el Basic.

Es el lenguaje más rápido que puede encontrarse para un microordenador, ya que puede suministrar directamente al microprocesador los códigos de las instrucciones a ejecutar, sin pasar por una interfase software como el interpretador del Basic. La diferencia de velocidad entre estos dos lenguajes es considerable (el lenguaje máquina va alrededor de 100 veces más rápido). La diferencia es menos acentuada con los lenguajes compilados, como son el Pascal o el Basic compilado. Estos son más rápidos que los lenguajes interpretados, porque el programa escrito en lenguaje de alto nivel se traduce una sola vez a lenguaje máquina por un procedimiento llamado «compilación». El programa traducido podrá ser ejecutado directamente por el microprocesador. Por el contrario, los interpretadores no traducen las instrucciones del lenguaje de alto nivel, sino más bien simulan su funcionamiento durante la ejecución, haciendo uso de las posibilidades del microprocesador. Si los compiladores fuesen perfectos producirían el código máquina más rápido posible para un programa determinado en lenguaje de alto nivel. En este caso sería inútil programar directamente en lenguaje máquina. De hecho, los compiladores producen un código máquina poco optimizado, que es varias veces más lento que el mismo programa realizado directamente en lenguaje máquina y que utiliza mejor las posibilidades del microprocesador.

Además de la rapidez, el lenguaje máquina permite acceder a todas las posibilidades del microordenador y escapar así de las restricciones impuestas por el lenguaje de alto nivel. Con el ensamblador se podrá programar la salida sonora para producir algo distinto al «bip» clásico, utilizar las dos últimas líneas de la pantalla para un fin distinto al de recoger información debido a la orden de INPUT, o utilizar una cantidad mayor de caracteres gráficos programados.

Si el lenguaje máquina solamente presentara estas ventajas, nos preguntaríamos porqué los constructores de ordenadores se empeñan en implantar el Basic en la versión de base de su microordenador. De hecho, el lenguaje máquina es mucho más difícil de utilizar para un neófito. El mínimo programa de cálculo aritmético con registro de resultados sobre la pantalla que se escribe en unas pocas líneas Basic, necesita varios centenares de instrucciones en lenguaje máquina, simplemente porque el microprocesador no sabe hacer otra cosa que sumas, restas y operaciones lógicas sobre números enteros codificados sobre 8 o 16 bits.



### 1.3. El ensamblador

La programación en código máquina es dura y ardua; debido a ello se ha creado el lenguaje ensamblador que asocia a cada instrucción de máquina una sucesión de caracteres alfanuméricos formando una palabra que recuerda de forma mnemotécnica la operación realizada por la instrucción. Se le llama *palabra mnemotécnica*.

Pongamos por ejemplo la operación: negación (cambio de signo). El código de esta instrucción es:

ED 44 (consta de dos octetos)

Su mnemotécnico es:

NEG

Para el programador, el mnemotécnico NEG es mucho más fácil de retener que el código hexadecimal ED 44, mucho más si tenemos en cuenta que hay centenares de instrucciones en lenguaje máquina (696 instrucciones en el Z 80).

Los mnemotécnicos no son directamente ejecutables por el microprocesador. Se ha debido crear una interfase software que asegure la transcripción de los mnemotécnicos a códigos máquina (ensamblaje). Esta interfase es un programa generalmente escrito en lenguaje máquina al que se le llama *Ensamblador*. Además de la transcripción de mnemotécnicos a códigos máquina, el ensamblador aporta facilidades para la edición de programas en lenguaje ensamblador, facilidades para manipular las variables y los datos, así como la posibilidad de especificar las direcciones de bifurcación en las instrucciones de salto.

El hecho de que la puesta a punto de los programas en lenguaje máquina suele ser más difícil que la de los programas en Basic, ha motivado la creación de programas potentes para la puesta a punto. A estos programas se les llama en inglés *Debugger*. Generalmente son capaces de efectuar el desensamblaje de las instrucciones correspondientes (operación inversa del ensamblaje). A menudo poseen una orden para ejecutar las instrucciones paso a paso, lo que es muy práctico para la puesta a punto y para la comprensión del funcionamiento de las instrucciones por el principiante.

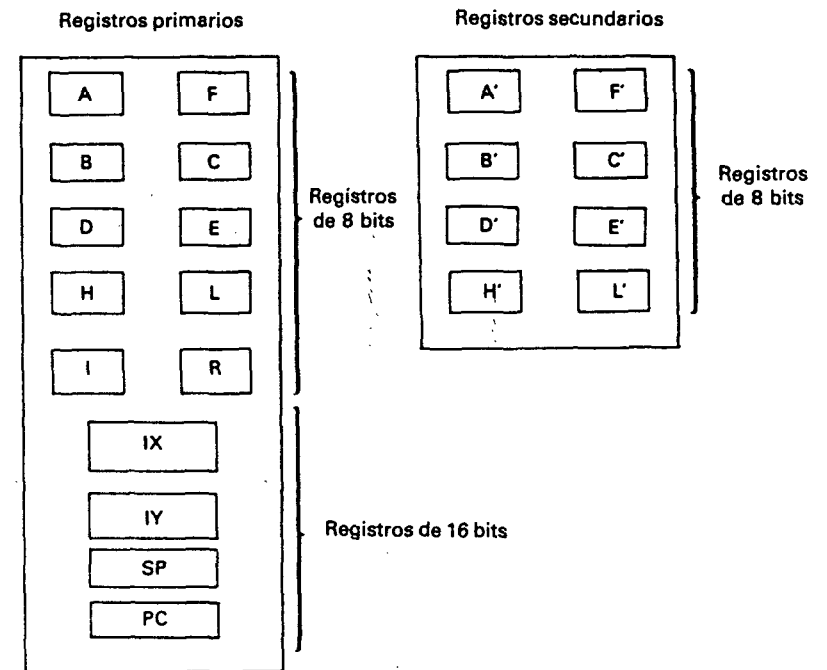
### 1.4. Los registros del Z 80

Para trabajar, el microprocesador dispone de una cantidad de registros, que son memorias de trabajo ultrarrápidas, situadas en su interior. El Z 80 posee instrucciones para hacer operaciones en los re-

gistros y otras para realizar transferencias entre registros o bien entre registro y la memoria central. A *grosso modo*, un programa en lenguaje máquina utilizará la siguiente estructura:

- Transferencia de datos desde la memoria viva hacia los registros.
- Manipulación de los datos memorizados en los registros.
- Transferencia de los resultados contenidos en los registros hacia la memoria central.

El Z 80 contiene en total 22 registros y cada uno tiene utilizaciones particulares.



Los registros A, B, C, D, E, H y L son registros de 8 bits para usos generales. Se utilizan para manipular o memorizar temporalmente un dato representado por un octeto.

El registro A tiene una función particular. Todas las operaciones lógicas o aritméticas de 8 bits se efectuarán entre el registro A y otro registro o un octeto de la memoria. El resultado siempre se coloca en el registro A que ha recibido el nombre de *acumulador*.

El registro B a menudo se utiliza como contador de bucle en un programa. La instrucción DJNZ, que se parece a la orden NEXT del Basic, utiliza este registro como variable de bucle.

Los registros B y C, que son dos registros de 8 bits, pueden agruparse para formar un registro de 16 bits al que se llama BC. El registro B contiene los 8 bits más significativos del registro BC.

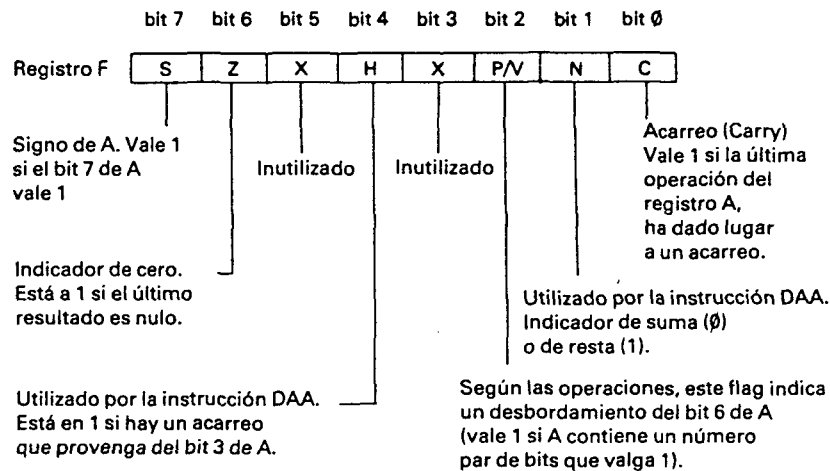


Lo mismo ocurre con los registros D,E y H,L que forman respectivamente los registros dobles DE y HL.

El registro HL trabaja como acumulador para las operaciones aritméticas. Las sumas y restas se realizan entre este registro y uno de los registros BC o DE. El resultado se coloca en HL.

Observemos que por este procedimiento de agrupación de registros, el microprocesador Z 80 puede manipular registros de 16 bits. El Z 80, de hecho, es un pseudomicroprocesador de 16 bits con bus de datos de 8 bits solamente. Esta característica lo hace más potente frente a los otros microprocesadores de 8 bits.

El registro F tiene 6 indicadores o banderas (flags en inglés). Cada indicador está hecho de un bit que se encuentra o bien en el estado VERDADERO (1), o bien en el estado FALSO (0). El cuadro siguiente resume los flags del registro F.



La mayor parte de estos indicadores informan al programa del estado del registro A después de una operación lógica o aritmética. Las instrucciones de salto condicional permiten comprobar el estado de un flag y bifurcar o no a una dirección de la memoria central según el estado del flag. Estas instrucciones de salto condicional asociadas al registro F, son el equivalente de la orden IF ... THEN ... del Basic.

Ciertas instrucciones modifican los flags según criterios distintos del estado del registro A. Otras no las modifican, pero sí provocan una modificación del registro A. Así pues, al usuario le corresponde asegurar que para cada una de las comprobaciones que efectúe, los flags contengan exactamente el valor deseado. Esta es una de las desventajas del lenguaje máquina, ya que es fuente de numerosos errores para los principiantes.

En la práctica se comprueba que, esencialmente, se utilizan los flags de cero (Z) y acarreo (C de carry) y menos a menudo el indicador de signo (S). Veremos con más detalle la utilización de los flags cuando examinemos el juego de instrucciones.

Los registros secundarios (A', F', B', C', D', E', H', L') funcionan de la misma forma que sus homólogos primarios (A, F, B, C, D, E, H, L). Estos permiten aumentar la capacidad de memorización del microprocesador. No obstante, tienen un interés limitado por el hecho de que el microprocesador no puede manipular simultáneamente los registros primarios y los registros secundarios. El Z 80 no puede llegar más que a los registros primarios, pero posee dos instrucciones (EX, AF, AF' y EXX) para cambiar de un bloque dos registros primarios con los registros secundarios.

El registro I, que estudiaremos con más detalle a continuación, se utiliza exclusivamente para gestionar las interrupciones.

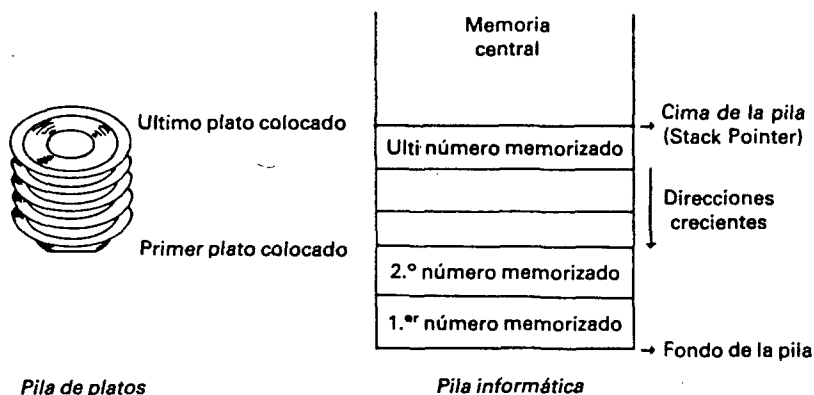
El registro R es un contador utilizado para refrescar las memorias dinámicas. Está controlado por el Z 80 y no es utilizable como registro por el usuario. El refresco de las memorias consiste en reescribir periódicamente el contenido de las memorias, ya que tienden a perder la información que contienen a causa de su tecnología dinámica. El registro R, utilizado como contador, permite saber cuándo debe ser ordenado el proceso de refresco. Su valor es aleatorio y no podrá ser utilizado más que para esto (generador de números aleatorios).

Los cuatro últimos registros (IX, IY, SP y PC), son registros de 16 bits que no pueden descomponerse en dos registros de 8 bits.

Los registros IX e IY son registros de índice. Son utilizados para permitir el direccionado por índice que estudiaremos en el párrafo siguiente. No obstante, pueden ser empleados como registros clásicos de 16 bits para memorizar un valor o realizar operaciones aritméticas.

El registro SP es el puntero de la pila (abreviación de stack pointer). La pila es un artefacto software creado para poder utilizar subprogramas y para la salvaguarda temporal de registros. Su funcionamiento es análogo al de una pila de platos. En una pila de platos solamente puede retirarse fácilmente el plato de encima y no puede colocarse otro plato más que encima de la misma. Esta pila sigue la regla de «último en llegar, primero en salir» (en inglés LIFO: last in, first out). El último plato colocado es el primero en retirarse.

En informática la pila (stack) es parecida a una pila donde los platos son sustituidos por números de 16 bits. El lugar donde van a ser colocados estos números es la memoria central. La cima de la pila está simbolizada por el registro SP que contiene la dirección de memoria del último número registrado.

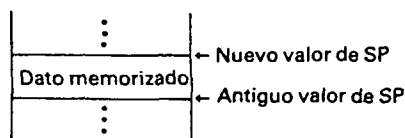


Pila de platos

Pila informática

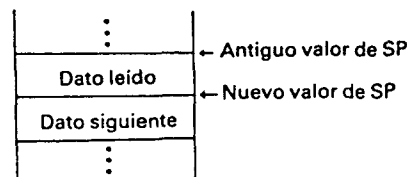
La operación de colocar un dato sobre la cima del stack se llama *apilamiento*. Esto se resume por las operaciones siguientes:

- $SP \leftarrow SP - 2$ : decrementar el stack pointer de pila en 2.
- $(SP) \leftarrow \text{dato para apilar}$ : colocar el dato en la posición de memoria cuya dirección está en SP. Se dice que SP apunta hacia esta posición de memoria.



A la operación de retirar el dato situado en la cima de la pila se le llama *desapilamiento*. Se resume por las operaciones siguientes:

- $\text{Dato leído} \leftarrow (SP)$ : leer el dato cuya dirección está situada en SP.
- $SP \leftarrow SP + 2$ : incrementar en 2 el stack pointer.



La pila puede utilizarse para salvaguardar temporalmente algunos datos en la memoria central. Para ello se apilan sucesivamente todos los datos que se desea resguardar:

- Apilar dato 1
- Apilar dato 2
- Apilar dato 3

Cuando llega el momento de recuperar estos datos, se efectúan tantos desapilamientos como apilamientos fueron realizados, pero en sentido inverso, pues debe tenerse en cuenta la regla: «último en llegar, primero en salir».

- Desapilar dato 3
- Desapilar dato 2
- Desapilar dato 1

El stack también sirve para las llamadas y retornos de los subprogramas. En el momento de una llamada de subprograma, la dirección de retorno se apila, es decir, la dirección que sigue a la llamada del subprograma. En el momento de un retorno de subprograma se desapila la dirección de retorno y se bifurca a esta dirección.

Gracias a esta estructura de stack pueden efectuarse llamadas a subprogramas imbricados (intercalados). La orden de retorno terminará el programa más interior.

En la utilización simultánea del stack por los subprogramas, y para la salvaguarda temporal de datos, habrá que asegurarse de que en el subprograma haya tantos apilamientos de datos como desapilamientos. En caso contrario, el dato presente en la cima de la pila (stack) en el momento del retorno del subprograma no será la dirección de retorno anteriormente apilada, lo que creará el riesgo de un bloqueo de programa.

El último registro del microprocesador Z 80 es el contador ordinal (PC: abreviación de Program Counter) que ya hemos visto.

## 1.5. Modos de direccionado

Los modos de direccionado describen cómo deben tomarse los datos utilizados por una instrucción. Para mejor comprender este concepto tomaremos como ejemplo la instrucción de transferencia. Esta instrucción ha recibido el mnemotécnico LD (abreviación de LOAD: carga en inglés). La instrucción completa se escribe:

LD x, y

Su efecto es transferir el valor de y a x. Es el LET  $x = y$  del Basic, donde los operandos (x e y) son enteros de 8 o 16 bits, y es el operando *fuer*te, x es el operando *destino*.

### 1.5.1. Direccionado por registro

En este tipo de direccionado, el operando es un registro de 8 o 16 bits.

*Ejemplo:*

78	LD	A, B	transfiere el contenido del
código	mnemónico	operandos	registro B al registro A

En este ejemplo los operandos fuente y destino son, los dos, registros de 8 bits.

### 1.5.2. Direccionado inmediato

El operando es una constante de 8 o 16 bits que se memoriza en 1 o 2 octetos de la instrucción.

*Ejemplos:*

3E 05 LD A,5

Esta instrucción transfiere el valor 5 al registro A. El valor 5 es memorizado en el segundo octeto de la instrucción. El operando fuente se obtiene por direccionado inmediato, mientras que el operando destino se obtiene por direccionado de registro. Este último no puede ser del tipo direccionado inmediato ya que debe ser modificado.

01 3A 1E LD BC,1E3AH

Esta instrucción transfiere el valor hexadecimal 1E3A (la H en la instrucción indica un valor hexadecimal), al registro doble BC. El dato de 16 bits 1E3A se guarda en el segundo y tercer octeto de la instrucción. El orden de almacenaje en memoria de los dos octetos que contengan el dato es un poco desanimador para un principiante. El segundo octeto 3A (*el octeto menos significativo u octeto de menos peso*) se almacena el primero y en la dirección N + 1. El primer octeto 1E (*el octeto más significativo o de más peso*) se almacena en segundo lugar en la dirección N + 2. El código de la instrucción (01) se almacena en la dirección N.

### 1.5.3. Direccionado directo

En esta forma de direccionado se suministra la dirección de memoria donde se encuentra el operando.

*Ejemplos:*

3A ED 59 LD A,(59EDH)

Esta instrucción transfiere el dato de 8 bits situado en la dirección 59EDH al registro A. La dirección es memorizada en los octetos 2 y 3 de la instrucción. Los paréntesis simbolizan el contenido de la dirección.

2A ED 59 LD HL,(59EDH)

Esta otra instrucción transfiere el dato de 16 bits situado en la dirección 59EDH, al registro doble HL. El octeto situado en la dirección 59EDH se coloca en L y el situado en la dirección 59EEH (dirección siguiente) se coloca en H. De aquí sale la regla de memorización de los números de 16 bits.

22 ED 59 LD (59EDH), HL

Esta última instrucción realiza la transferencia inversa de la precedente.

### 1.5.4. Direccionado indirecto por registro

Para esta forma se suministra el registro doble que contiene la dirección del operando de 8 bits.

*Ejemplo:*

7E LD A,(HL)

El dato situado en la dirección que está contenida en HL es transferido a A. Se dice que HL *apunta* hacia este dato. HL es calificado de *indicador o puntero*.

Los dos grupos de instrucciones siguientes transfieren el mismo valor a A:

LD HL,59EDH	LD A,(59EDH)
LD A,(HL)	

### 1.5.5. Direccionado por índice

Este método es una extensión del precedente. La dirección del operando es igual al contenido de uno de los registros del índice (IX o IY) más un valor de 8 bits que se le proporciona.

*Ejemplo:*

```
DD 46 09    LD B,(IX + 9)
```

Esta instrucción calcula la dirección del operando añadiéndole 9 al valor contenido en IX. El octeto situado en esta dirección se coloca en el registro B. El valor del desplazamiento es almacenado en el tercer octeto de la instrucción. Este desplazamiento puede ser positivo o negativo. Si el valor del octeto que lo representa es superior a 7FH(127), el desplazamiento será negativo e igual en valor absoluto a 256 menos el valor del octeto. Así, el octeto de valor F7 representa un desplazamiento de -9. Entonces la instrucción se escribe:

```
DD 46 F7    LD B,(IX - 9)
```

Este procedimiento de codificación de números negativos es conocido como codificación *en complemento a 2*. En esta codificación, un número negativo se caracteriza por la puesta a uno del bit más significativo (bit 7 para un número de 8 bits).

### 1.5.6. Direccionado relativo

Esta clase de direccionado se utiliza con la instrucción de salto relativo JR que provoca un desplazamiento positivo o negativo del contador ordinal.

*Ejemplo:*

```
18 03    JR $ + 5
```

Esta instrucción provoca un salto de 5 octetos hacia adelante. Se parece a la instrucción GOTO del Basic, excepto en que el punto de bifurcación se da relativo a la posición corriente, y no de forma absoluta como en el Basic. La expresión \$ + 5 indica la dirección de la bifurcación (\$ es el valor del contador ordinal al principio de la instrucción). El desplazamiento se almacena en el segundo octeto de la instrucción.

Utilizando la codificación en complemento a dos puede especificarse un desplazamiento negativo.

*Ejemplo:*

```
18 F9    JR $ - 5
```

### 1.5.7. Direccionado por bit

Algunas instrucciones permiten manipular directamente los bits de un número de 8 bits. SET pone un bit a uno, RES coloca un bit a cero y BIT comprueba el valor de un bit, y posiciona el indicador Z en consecuencia. Los bits son representados por un número del 0 al 7. El bit 0 es el menos significativo y el bit 7 es el más significativo.

*Ejemplos:*

CB DF	SET 3, A	puesta a 1 del bit 3 de A
CB AE	RES 5, (HL)	puesta a 0 del bit 5 de (HL)
DD CB 03 46	BIT 0, (IX + 3)	comprueba el bit 0 de (IX + 3)

## 1.6. Juego de instrucciones del Z 80

En este apartado estudiaremos todas las instrucciones disponibles en el Z 80. El número de instrucciones es relativamente grande para un microprocesador de 8 bits (696 instrucciones). Este es uno de los puntos fuertes del Z 80.

Para cada instrucción daremos el detalle de la operación efectuada, el detalle de la eventual modificación de los indicadores del registro F y el número de ciclos de reloj utilizados para realizar la instrucción. Esta cantidad de ciclos permite calcular explícitamente la duración de una instrucción. Sabiendo que la frecuencia de reloj es de 3,25 MHz el tiempo de ciclo es igual a 0,3 µs. La duración de una instrucción es igual al producto del número de ciclos por la duración del ciclo del reloj. Una instrucción simple necesita cuatro ciclos y dura como consecuencia 1,2 µs, o sea un poco más de una millonésima de segundo (el microsegundo, de símbolo µs, es la millonésima parte de un segundo). Percibimos así la extrema rapidez del lenguaje máquina en comparación con el Basic.

### 1.6.1. Transferencia de 8 bits

La instrucción LD d,s que ya hemos visto, realiza la transferencia del operando s al operando d. Estos operandos se describen en uno de los modos de direccionado del Z 80. No obstante, todas las combinaciones de modos de direccionado no están permitidas para s y d. Examinemos la lista de instrucciones de este tipo que están permitidas:

Mnemotécnicos	Operación	Código de la instrucción binaria	Hexadecimal	Número de ciclos de reloj	Indicadores
LD r, s	r ← s	0 1 ← r → ← s →		4	no se modifican
LD r, n	r ← n	0 0 ← r → 1 1 0 ← n →		7	no se modifican
LD r, (HL)	r ← (HL)	0 1 ← r → 1 1 0		7	no se modifican
LD r, (IX+d)	r ← (IX+d)	1 1 0 1 1 1 0 1 0 1 ← r → 1 1 0 ← d →	DD	19	no se modifican
LD r, (IY+d)	r ← (IY+d)	1 1 1 1 1 1 0 1 0 1 ← r → 1 1 0 ← d →	FD	19	no se modifican
LD (HL), r	(HL) ← r	0 1 1 1 0 ← r →		7	no se modifican
LD (IX+d), r	(IX+d) ← r	1 1 0 1 1 1 0 1 0 1 1 1 0 ← r → ← d →	DD	19	no se modifican
LD (IY+d), r	(IY+d) ← r	1 1 1 1 1 1 0 1 0 1 1 1 0 ← r → ← d →	FD	19	no se modifican
LD (HL), n	(HL) ← n	0 0 1 1 0 1 1 0 ← n →	36	10	no se modifican
LD (IX+d), n	(IX+d) ← n	1 1 0 1 1 1 0 1 0 0 1 1 0 1 1 0 ← d → ← n →	DD 36	19	no se modifican
LD (IY+d), n	(IY+d) ← n	1 1 1 1 1 1 0 1 0 0 1 1 0 1 1 0 ← d → ← n →	FD 36	19	no se modifican
LD A, (BC)	A ← (BC)	0 0 0 0 1 0 1 0	0A	7	no se modifican
LD A, (DE)	A ← (DE)	0 0 0 1 1 0 1 0	1A	7	no se modifican

LD A, (nn)	A ← (nn)	0 0 1 1 1 0 1 0 ← n: menos peso → ← n: más peso →	3A	13	no se modifican
LD (BC), A	(BC) ← A	0 0 0 0 0 1 0	02	7	no se modifican
LD (DE), A	(DE) ← A	0 0 0 1 0 0 1 0	12	7	no se modifican
LD (nn), A	(nn) ← A	0 0 1 1 0 0 1 0 ← n: menos peso → ← n: más peso →	32	13	no se modifican
LD I, A	I ← A	1 1 1 0 1 1 0 1 0 1 0 0 0 1 1 1	ED 47	9	no se modifican
LD R, A	R ← A	1 1 1 0 1 1 0 1 0 1 0 0 1 1 1 1	ED 4F	9	no se modifican
LD A, I	A ← I	1 1 1 0 1 1 0 1 0 1 0 1 0 1 1 1	ED 57	9	S y Z posicionados según el estado de A; H y N puestos a 0; C sin afectar; P/V contienen el estado de biestable de interrupciones. Indicadores modificados como en el caso precedente.
LD A, R	A ← R	1 1 1 0 1 1 0 1 0 1 0 1 1 1 1 1	ED 5F	9	

En este cuadro *n* designa una constante de 8 bits, *d* un desplazamiento de 8 bits positivo o negativo (direccionado por índice), *nn* una constante de 16 bits y *r* o *s* un registro de 8 bits. Estos registros son codificados por el Z 80 sobre tres bits que también se encuentran en el código de la instrucción. La correspondencia entre los registros y estos tres bits se resume en el cuadro siguiente.

Registro	ros
B	000
C	001
D	010
E	011
H	100
L	101
A	111

Una lista exhaustiva de estas instrucciones se encuentra en el anexo.

Estas instrucciones de transferencia, que son fáciles de comprender, de hecho son las más empleadas en un programa. Según la forma de direccionado utilizada por los operandos, el número de octetos de la instrucción, así como la duración de la instrucción, varían. Estas instrucciones, salvo las dos últimas, no afectan a los indicadores.

### 1.6.2. Transferencia de 16 bits

Examinemos ahora la lista menos importante de las instrucciones de transferencia de 16 bits.

Mnemotécnico	Operación	Código de la instrucción binaria	Hexadecimal	Número de ciclos de reloj	Indicadores
LD dd, nn	dd ← nn	00 ← d d → 0001 ← n: menos peso → ← n: más peso →		10	
LD IX, nn	IX ← nn	11011101 00100001 ← n: menos peso → ← n: más peso →	DD 21	14	no se modifican
LD IY, nn	IY ← nn	11111101 00100001 ← n: menos peso → ← n: más peso →	FD 21	14	no se modifican
LD HL, (nn)	H ← (nn+1) L ← (nn)	00101010 ← n: menos peso → ← n: más peso →	2A	16	no se modifican
LD dd, (nn)	dd ← (nn)	11101101 01 ← d d → 1011 ← n: menos peso → ← n: más peso →	ED	20	no se modifican
LD IX, (nn)	IX ← (nn)	11011101 00101010 ← n: menos peso → ← n: más peso →	DD 2A	20	no se modifican

LD IY, (nn)	IY ← (nn)	11111101 00101010 ← n: menos peso → ← n: más peso →	FD 2A	20	no se modifican
LD (nn), HL	(nn+1) ← H (nn) ← L	00100010 ← n: menos peso → ← n: más peso →	22	16	no se modifican
LD (nn), dd	(nn) ← dd	11101101 01 ← d d → 0011 ← n: menos peso → ← n: más peso →	ED	20	no se modifican
LD (nn), IX	(nn) ← IX	11011101 00100010 ← n: menos peso → ← n: más peso →	DD 22	20	no se modifican
LD (nn), IY	(nn) ← IY	11111101 00100010 ← n: menos peso → ← n: más peso →	FD 22	20	no se modifican
LD SP, HL	SP ← HL	11111001	F9	6	no se modifican
LD SP, IX	SP ← IX	11011101 11111001	DD F9	10	no se modifican
LD SP, IY	SP ← IY	11111101 11111001	FD F9	10	no se modifican

En este cuadro *nn* designa un número de 16 bits y *dd* uno de los registros de 16 bits: BC, DE, HL, SP. Estos registros se codifican sobre dos bits que se encuentran en el código de la instrucción.

Registro	dd
BC	00
DE	01
HL	10
SP	11

Los modos de direccionado posibles, con las instrucciones de transferencia de 16 bits, son menos importantes que los disponibles en las instrucciones de transferencia de 8 bits. No encontramos más que el modo directo y el modo por registro.

Estas instrucciones son muy útiles para manipular los datos de 16 bits o las direcciones de memoria codificadas en 16 bits. Se las emplea en particular para iniciar los registros dobles en vistas a utilizar el modo de direccionado indirecto sobre los números de 8 bits.

### 1.6.3. Operaciones aritméticas de 8 bits

El Z80 es capaz de efectuar operaciones aritméticas simples sobre números de 8 bits, codificados o no, en complemento a 2. Este tipo de codificación, que ya hemos visto, permite representar en un octeto todos los números enteros comprendidos entre  $-128$  y  $+127$ . Los números positivos se codifican normalmente mientras que los números negativos son representados por un número igual a 256 menos su valor. Así, los números de  $-128$  a  $-1$  se representan por los números de 128 a 255 (representación normal).

Valor del octeto en representación normal	0	1	...	126	127	128	129	...	254	255
Valor del octeto en complemento a dos	0	1	...	126	127	-128	-127	...	-2	-1

Con esta codificación, un número negativo se reconoce por la presencia del valor 1 en el bit más significativo del octeto. Este es el bit que comprueba el indicador de signo S en el momento de las operaciones aritméticas.

De hecho, el microprocesador no hace diferencia entre estos dos tipos de codificaciones. Todas las operaciones aritméticas se realizan sobre la representación normal de números, con pérdida del acarreo o sin ella, en caso de desbordamiento. Esta posibilidad de acarreo asegura la equivalencia de los cálculos para los dos tipos de representación.

*Ejemplos:* Tomemos por caso la suma de 12 con  $-4$ .  $-4$  es codificado por  $256 - 4 = 252$  en representación normal. La suma sería en binario:

	Representación normal	Representación en complemento a 2
$\emptyset \emptyset \emptyset \emptyset 1 1 \emptyset \emptyset$	12	12
+ $1 1 1 1 1 1 \emptyset \emptyset$	252	-4
<hr/>		
$\textcircled{1} \emptyset \emptyset \emptyset \emptyset 1 \emptyset \emptyset \emptyset$	264	8

C ← Los 8 bits resultantes puestos en A

El número resultante contiene 9 bits. Sólo los 8 bits menos significativos se conservan. El último bit se coloca en el indicador de acarreo C. El número resultante es pues igual a 8 si no se tiene en cuenta el acarreo (cálculo en complemento a 2) y a  $256 + 8 = 264$ , si se tiene en cuenta (cálculo en representación normal:  $12 + 252 = 264$ ).

Corresponde al usuario el tener o no tener en cuenta la indicación del acarreo, según el tipo de datos que manipule.

Tomemos como otro ejemplo la suma de  $-1$  y de  $-2$  que vale  $-3$ . Esta operación sería en representación normal:

$$256 - 1 + 256 - 2 = \underbrace{256}_{\text{acarreo}} + \underbrace{256 - 3}_{\text{valor sobre 8 bits}}$$

El resultado de 8 bits vale  $256 - 3$ , o sea el valor  $-3$  en complemento a dos, y el indicador C indica un acarreo.

Si el usuario trabaja en representación normal deberá sumar los números 255 y 254 y obtendrá el resultado de 256 (acarreo a 1) más 253, o sea 509 que es la suma de 255 y de 254.

En la práctica se utilizará la codificación en complemento a dos para números comprendidos entre  $-128$  y  $+127$  y la codificación en representación normal para números positivos pudiendo sobrepasar el valor 127.

El calificativo de complemento a dos de la codificación proviene del hecho de que la representación normal de un número negativo puede obtenerse haciendo el *complemento* de cada uno de los bits del número positivo correspondiente y añadiendo uno al resultado. La operación de complemento consiste en cambiar los  $\emptyset$  por 1 y los 1 por  $\emptyset$ .

*Ejemplo:*

3 se escribe en binario :  $\emptyset \emptyset \emptyset \emptyset \emptyset \emptyset 1 1$   
 El complemento de 3 vale :  $1 1 1 1 1 1 \emptyset \emptyset$   
 Añadamos 1 al resultado :  $1 1 1 1 1 1 \emptyset 1$

El resultado obtenido es  $256 - 3$  en representación de complemento a 2.



El cuadro siguiente describe las instrucciones de suma y resta sobre 8 bits.

Mnemotécnico	Operación	Código interno	Indicadores
ADD A, s	$A \leftarrow A + s$	0 0 0	S puesto a 1 si hay resultado negativo, si no puesto a 0. Z puesto a 1 si hay resultado nulo, si no puesto a 0.
ADCA, s	$A \leftarrow A + s + \text{Carry}$	0 0 1	H puesto a 1 si hay acarreo del bit 3, si no a 0. P/V puesto a 1 si hay acarreo del bit 6, si no puesto a 0.
SUB s	$A \leftarrow A - s$	0 1 0	N puesto a 0 para una suma y a 1 para una resta.
SBC A, s	$A \leftarrow A - s - \text{Carry}$	0 1 1	C puesto a 1 si hay acarreo del bit 7, si no puesto a 0.

Todas estas operaciones aritméticas se realizan entre el registro A y el dato de 8 bits s; el resultado se coloca en A con posicionamiento de los indicadores según el resultado.

El símbolo s designa uno de los cinco modos de direccionado indicados en el siguiente cuadro; el código completo de la instrucción se obtiene por inserción del código interno en uno de los códigos siguientes:

Código de la instrucción	s	Modo de direccionado	Número de ciclos
Registro	r	1 0 <span style="border: 1px solid black; padding: 2px;">código interno</span> ← r →	4
Dato de 8 bits	n	1 1 <span style="border: 1px solid black; padding: 2px;">código interno</span> 1 1 0 ← n →	7
Indirecto HL	(HL)	1 0 <span style="border: 1px solid black; padding: 2px;">código interno</span> 1 1 0	7
Por índice IX	(IX + d)	1 1 0 1 1 1 0 1 DD 1 0 <span style="border: 1px solid black; padding: 2px;">código interno</span> 1 1 0 ← d →	19
Por índice IY	(IY + d)	1 1 1 1 1 1 0 1 FD 1 0 <span style="border: 1px solid black; padding: 2px;">código interno</span> 1 1 0 ← d →	19

El indicador P/V permite saber si ha habido un acarreo del bit 6 de A; dicho en otros términos, si ha habido un desbordamiento en los cálculos para números codificados en complemento a dos:

Ejemplo:  $100 + 84 = 128 + 56$

bit de desbordamiento	resto sobre 7 bits
0 1 1 0 0 1 0 0	1 0 0
+ 0 1 0 1 0 1 0 0	84
-----	
1 0 1 1 1 0 0 0	184
desbordamiento	resultado sobre 7 bits

El indicador P/V permite saber que en este caso el resultado no es -72 (184 en representación normal) sino  $128 + 56$ .

Por lo tanto, si se trabaja con números codificados en complemento a dos, es necesario fijarse en el indicador P/V, y no en el indicador Carry utilizado en la representación normal.

Otras dos instrucciones utilizadas frecuentemente permiten el incremento o decremento del operador en una unidad.

Mnemotécnico	Operación	Indicadores
INC s	$s \leftarrow s + 1$	Los indicadores S, Z, H y P/V son modificados de la misma forma que en las otras instrucciones aritméticas. N es puesto a 1 para la instrucción INC y a 0 para DEC. C no se modifica.
DEC s	$s \leftarrow s - 1$	

El cuadro siguiente resume los modos de direccionado posibles:

Modo de direccionado	s	Código de la instrucción	Número de ciclos
Registro	r	0 0 ← r → 1 0 X	4
Indirecto HL	(HL)	0 0 1 1 0 1 0 X	11
Por índice IX	(IX + d)	1 1 0 1 1 1 0 1 DD 0 0 1 1 0 1 0 X ← d →	23
Por índice IY	(IY + d)	1 1 1 1 1 1 0 1 FD 0 0 1 1 0 1 0 X ← d →	23

El bit X vale 0 para la instrucción INC y 1 para la instrucción DEC. Finalmente, la instrucción CP permite comparar el valor del acumulador con el operando. Esta instrucción efectúa la sustracción  $A - s$  y proporciona los indicadores según el resultado. Este último no se copia en el registro A que no sufre modificación.

Mnemotécnico	Operación	Código interno
CP s	test $A - s$	1 1 1

Los modos de direccionado son los mismos que para las operaciones aritméticas ADD y SUB.

*Ejemplo:*

CP 30

Esta instrucción compara el acumulador con el valor 30.

Z = 1 si  $A = 30$   
 C = 1 si  $A < 30$   
 C = 0 si  $A \geq 30$

Utilizando las informaciones suministradas por los indicadores Z y C, un valor de 8 bits contenido en el registro A puede compararse con el valor de 8 bits representado por el operando.

#### 1.6.4. Operaciones lógicas de 8 bits

Las instrucciones lógicas efectúan una operación lógica entre cada uno de los bits del acumulador y cada uno de los bits del operando. Es posible efectuar tres operaciones lógicas diferentes entre dos bits:

El operador AND hace el «Y lógico» entre los dos bits operandos. El resultado solamente es 1 si el primero y el segundo operando están a 1.

El operador OR hace la «O lógica» entre los dos operandos. El resultado es 1 si el primero o el segundo operando están a 1.

El operador XOR hace la «O exclusiva» entre los dos operandos. El resultado será 1 si el primero o el segundo operando están a 1, pero no los dos al mismo tiempo.

La tabla de la verdad siguiente resume estas funciones:

Primer operando	Segundo operando	AND	OR	XOR
0	0	0	0	0
0	1	0	1	1
1	0	0	1	1
1	1	1	1	0

Las instrucciones lógicas correspondientes del Z 80 efectúan estas operaciones en paralelo sobre los 8 bits del acumulador y del operando, y dejan el resultado en el acumulador.

Mnemotécnico	Operación	Código interno	Indicadores
AND s	$A \leftarrow A \text{ Y } s$	1 0 0	S puesto a 1 si hay resultado negativo. Z puesto a 1 si hay resultado nulo.
OR s	$A \leftarrow A \text{ O } s$	1 1 0	H puesto a 1 para AND y si no a 0. P/V contiene la paridad del resultado.
XOR s	$A \leftarrow A \text{ O } s$ exclusiva	1 0 1	N puesto a cero. C puesto a cero.

Los mandos de direccionado son los mismos que para las instrucciones ADD y SUB.

Estas instrucciones son de mucha utilidad para acceder a cierto número de bits del registro A.

La instrucción AND sirve para aislar cierto número de bits del registro A, colocando los otros bits a cero. Se realiza así una máscara para seleccionar estos bits.

*Ejemplo:*

AND 15 permite aislar los cuatro bits de menor peso de A.

Si A contiene un 39H, contendrá un 9 después de la ejecución de esta instrucción. Solamente los 4 bits de menor peso han sido guardados; los otros han sido puestos a cero.

La instrucción OR provoca la puesta a 1 de unos cuantos bits del acumulador.

*Ejemplo:*

OR F0H provoca la puesta a 1 de los 4 bits de más peso de A.

Si A contiene un 39H, contendrá un F9H después de la ejecución de esta instrucción.

La instrucción OR A permite probar si el acumulador es nulo sin modificar este último, lo cual será empleado a menudo. También provoca la puesta a cero del indicador Carry, lo que será útil para el empleo de ADC y SBC.

La instrucción XOR sirve para complementar cierto número de bits de A.

*Ejemplo:*

XOR F0H complementa los 4 bits de más peso de A.

Si en A hubiera un 39H, contendría un C9H después de la ejecución de esta instrucción.

La instrucción XOR A provoca en particular la puesta a cero de todos los bits de A, o sea la puesta a cero de A. Esto equivale a LD A,0, pero presenta la ventaja de no ocupar más que un octeto en lugar de dos y de posicionar los indicadores.

### 1.6.5. Decalados

El Z 80 es capaz de decalar hacia la derecha o hacia la izquierda un número de 8 bits. Esta operación consiste en desplazar cada uno de los bits del número hacia la derecha o hacia la izquierda. El bit sobrante del decalado se utiliza de diferentes maneras según la instrucción aplicada.

*Ejemplos:*

1 0 0 1 1 1 1 0 (9E) decalado a la izquierda: 0 0 1 1 1 0 0 (3C)

1 0 0 1 1 1 1 0 (9E) decalado a la derecha : 0 1 0 0 1 1 1 (4F)

El bit saliente es el bit 7 después del decalado a la izquierda y el bit 0 después del decalado a la derecha. Este bit puede colocarse en el Carry o reinsertarse en el bit opuesto del número para obtener un decalado circular.

Sin la pérdida del bit saliente, el decalado a la izquierda es equivalente a una multiplicación por dos y el decalado a la derecha a una división por dos. En efecto, esta operación decala las cifras de un número escrito en base dos, lo que se convierte en una multiplicación o división del número por la base, que vale 2. Utilizaremos esta propiedad en la realización de subprogramas de multiplicación y de división.

Ahora examinemos el detalle de las instrucciones de decalado:

Mnemotécnico	Operación	Código	Número de ciclos	Indicadores
RLCA		0 0 0 0 0 1 1 1 0 7	4	S, Z y V sin modificar. H y N puesto a cero. C contiene el bit saliente.
RLA		0 0 0 1 0 1 1 1 1 7	4	
RRC A		0 0 0 0 1 1 1 1 0 F	4	
RRA		0 0 0 1 1 1 1 1 1 F	4	
RLD		1 1 1 0 1 1 0 1 ED 0 1 1 0 1 1 1 1 6F	18	S y Z posicionados según el valor resultante de A. H y N puestos a cero. C sin modificar. P/V contiene la paridad de A.
RRD		1 1 1 0 1 1 0 1 ED 0 1 1 0 0 1 1 1 67	18	

Las cuatro primeras instrucciones de este cuadro efectúan rotaciones en el acumulador. Las dos últimas son más especiales. Actúan sobre grupos de 4 bits contenidos en el registro A y en el octeto situado en la dirección HL. Sirven especialmente para la manipulación de los números codificados en BCD (código decimal binario) utilizados en aritmética decimal. Las cifras decimales son así codificadas sobre 4 bits. Así pues, un número decimal de dos cifras puede colocarse en un octeto.

*Ejemplo:*

72 decimal se escribe en BCD: 0 1 1 1 0 0 1 0 o sea 72H  
cifra 7 cifra 2

Las instrucciones RRD y RLD permiten realizar decalados de números decimales.

Las otras instrucciones de rotación con 8 bits se resumen en el siguiente cuadro:

Mnemotécnico	Operación	Código interno	Indicadores
RLC s		0 0 0	Z puesto a 1 si el resultado es nulo.
RL s		0 1 0	S puesto a 1 si el resultado es negativo.
RRC s		0 0 1	H puesto a 0.
RR s		0 1 1	P/V contiene la paridad del resultado.
SLA s		1 0 0	N puesto a 0.
SRA s		1 0 1	C contiene el bit saliente del decalado.
SRL s		1 1 1	

El signo s designa uno de los cuatro modos de direccionado en el cuadro siguiente; el código completo de la instrucción se obtiene entonces por inserción del código interno en uno de los códigos siguientes:

Modo de direccionado	s	Código	Números de ciclos
Registro	r	1 1 0 0 1 0 1 1 CB 0 0 código interno ← r →	8
Indirecto HL	(HL)	1 1 0 0 1 0 1 1 CB 0 0 código interno 1 1 0	15
Por índice IX	(IX + d)	1 1 0 1 1 1 0 1 DD 1 1 0 0 1 0 1 1 CB ← d → 0 0 código interno 1 1 0	23
Por índice IY	(IY + d)	1 1 1 1 1 1 0 1 FD 1 1 0 0 1 0 1 1 CB ← d → 0 0 código interno 1 1 0	23

La correspondencia entre el nombre del registro r y los 3 bits del código de la instrucción se da en el párrafo 1.6.1.

### 1.6.6. Instrucciones que actúan sobre un bit

El Z 80 posee tres instrucciones que permiten posicionar o comprobar un bit de un número de ocho bits.

Mnemotécnico	Operación	Código interno	Indicadores
BIT b, s	Comprobación del bit b de s.	0 1	Z está a 1 si el bit vale 0. H es puesto a 1, N es puesto a 0. S y P/V tienen un valor cualquiera. C no es modificado.
SET b, s	Puesta a 1 del bit b de s.	1 1	Los indicadores no son modificados.
RES b, s	Puesta a 0 del bit b de s.	1 0	

El número del bit  $b$  varía entre  $\emptyset$  y 7. Este número se encuentra en tres de los bits del código de la instrucción, según la correspondencia siguiente:

$b$	Código de 3 bits
$\emptyset$	$\emptyset \emptyset \emptyset$
1	$\emptyset \emptyset 1$
2	$\emptyset 1 \emptyset$
3	$\emptyset 1 1$
4	$1 \emptyset \emptyset$
5	$1 \emptyset 1$
6	$1 1 \emptyset$
7	$1 1 1$

La tabla siguiente resume los diferentes modos de direccionado posibles para  $s$ .

Modo de direccionado	$s$	Código de la instrucción	Número de ciclos
Registro	$r$	$1 1 \emptyset \emptyset 1 \emptyset 1 1$ CB 	8
Indirecto HL	(HL)	$1 1 \emptyset \emptyset 1 \emptyset 1 1$ CB 	12 para BIT 15 para SET y RES
Por índice IX	(IX + d)	$1 1 \emptyset 1 1 1 \emptyset 1$ DD $1 1 \emptyset \emptyset 1 \emptyset 1 1$ CB 	2 $\emptyset$ para BIT 23 para SET y RES
Por índice IY	(IY + d)	$1 1 1 1 1 1 \emptyset 1$ FD $1 1 \emptyset \emptyset 1 \emptyset 1 1$ CB 	2 $\emptyset$ para BIT 23 para SET y RES

### 1.6.7. Instrucciones de uso general

Mnemotécnico	Operación	Código de la instrucción	Número de ciclos	Indicadores
NOP	No hay operación	$\emptyset \emptyset \emptyset \emptyset \emptyset \emptyset \emptyset \emptyset \emptyset$	4	No se modifican.
SCF	Carry $\leftarrow 1$	$\emptyset \emptyset 1 1 \emptyset 1 1 1$ 37	4	C a 1, H y N a $\emptyset$ . Otros indicadores no se modifican.
CCF	Complemento del Carry	$\emptyset \emptyset 1 1 1 1 1 1$ 3F	4	H desconocido. C complementado, N a $\emptyset$ . Otros indicadores no se modifican.
CPL	Complemento de A $A \leftarrow \bar{A}$	$\emptyset \emptyset 1 \emptyset 1 1 1 1$ 2F	4	H y N a 1. Otros indicadores no se modifican.
NEG	Negación de A en complemento a 2 $A \leftarrow \bar{A} + 1$	$1 1 1 \emptyset 1 1 \emptyset 1$ ED $\emptyset 1 \emptyset \emptyset \emptyset 1 \emptyset \emptyset$ 44	8	N puesto a 1, los otros indicadores son posicionados según el resultado de A. P/V contiene el desbordamiento eventual del bit 6.
DAA	Ajuste decimal	$\emptyset \emptyset 1 \emptyset \emptyset 1 1 1$ 27	4	N no se modifica, P/V contiene la paridad del resultado. Los otros indicadores son posicionados según el resultado de A.
HALT	Paro del Z 8 $\emptyset$	$\emptyset 1 1 1 \emptyset 1 1 \emptyset$ 78	4	Indicadores sin modificar.

La instrucción NOP será de utilidad durante la puesta a punto. Gracias a ella podrá calcularse el efecto de una secuencia de instrucciones reemplazándolas por octetos nulos (instrucción NOP).

La instrucción DAA efectúa automáticamente el ajuste decimal que debe ser realizado a continuación de una suma o de una resta de dos números decimales codificados en BCD. Esta instrucción utiliza las informaciones facilitadas por los indicadores H y N.

*Ejemplo:* Consideremos la suma de 34 y 39:

```
LD A,34H ; A contiene 34 en BCD
ADD A,29H ; A contiene 5 DH que no es un número en BCD
DAA ; A contiene 63 en BCD que es el resultado de la suma
```

Después de una suma, la instrucción comprueba si la cifra de menos peso (4 bits de menor peso) es superior a 9, en cuyo caso se le añade 6 al número para obtener la nueva cifra de las unidades e incrementar la cifra de las decenas (4 bits de más peso). El mismo procedimiento sirve para la cifra de las decenas, pero añadiendo esta vez  $6 \cdot 16$  al número y posicionando el indicador Carry en caso de desbordamiento.

Esta instrucción, que es muy rápida, facilita considerablemente el empleo de los números decimales codificados en BCD. No obstante, tan sólo funciona con números de 8 bits, lo que hace más delicada la manipulación de números decimales de mayor tamaño.

### 1.6.8. Instrucciones aritméticas sobre 16 bits

Una de las ventajas del microprocesador Z 80 sobre sus predecesores es su capacidad para efectuar sumas y restas sobre números enteros de 16 bits. Las instrucciones se encuentran resumidas en el siguiente cuadro:

Mnemotécnico	Operación	Código de la instrucción	Número de ciclos	Indicadores
ADD HL, ss	$HL \leftarrow HL + ss$	$00 \leftarrow ss \rightarrow 1001$	11	N puesto a cero, H desconocido. S, Z, P/V no se modifican. C vale 1 si hay un acarreo del bit 15 de HL.
ADC HL, ss	$HL \leftarrow HL + ss + \text{Carry}$	$11101101 \text{ ED}$ $01 \leftarrow ss \rightarrow 1010$	15	N puesto a 0, H desconocido. S contiene el bit 15 de HL. Z vale 1 si HL es nulo. P/V vale 1 si hay un acarreo del bit 14. C vale 1 si hay un acarreo del bit 15.
SBC HL, ss	$HL \leftarrow HL - ss - \text{Carry}$	$11101101 \text{ ED}$ $01 \leftarrow ss \rightarrow 0010$	15	N puesto a 1. Los otros indicadores varían de la misma forma que en la instrucción ADC.

ADD IX, pp	$IX \leftarrow IX + pp$	$11011101 \text{ DD}$ $00 \leftarrow pp \rightarrow 1001$	15	N puesto a cero. H desconocido. S, Z, P/V no se modifican. C vale 1 si hay un acarreo del bit 15 de HL.	
ADD IY, rr	$IY \leftarrow IY + rr$	$11111101 \text{ FD}$ $00 \leftarrow rr \rightarrow 1001$	15		
INC ss	$ss \leftarrow ss + 1$	$00 \leftarrow ss \rightarrow 0011$	6		
INC IX	$IX \leftarrow IX + 1$	$11011101 \text{ DD}$ $00100011 \text{ 23}$	10		
INC IY	$IY \leftarrow IY + 1$	$11111101 \text{ FD}$ $00100011 \text{ 23}$	10		
DEC ss	$ss \leftarrow ss - 1$	$00 \leftarrow ss \rightarrow 1011$	6		
DEC IX	$IX \leftarrow IX - 1$	$11011101 \text{ DD}$ $00101011 \text{ 2B}$	10		
DEC IY	$IY \leftarrow IY - 1$	$11111101 \text{ FD}$ $00101011 \text{ 2B}$	10		
					No se modifican.

La notación en complemento a dos de números negativos es igualmente utilizable para números de 16 bits. Así se pueden codificar todos los números enteros comprendidos entre  $-32768$  y  $32767$ ; un número negativo representado por un número de 16 bits vale  $65536$  más el valor del número negativo. El bit 15 permite saber el signo del número (número negativo si el bit 15 es 1).

En estas operaciones aritméticas los indicadores comprueban el valor del resultado que es un número de 16 bits y ya no está en el registro A. El indicador Carry permitirá saber si hay un acarreo que provenga del bit 15 del resultado.

Los símbolos ss, pp y rr designan registros de 16 bits, resumidos en las tablas de más abajo, las cuales dan la correspondencia entre el nombre del registro y los dos bits que lo representan en el código de la instrucción.

Registro	ss
BC	00
DE	01
HL	10
SP	11

Registro	pp
BC	00
DE	01
IX	10
SP	11

Registro	rr
BC	00
DE	01
IY	10
SP	11

### 1.6.9. Instrucciones de apilamiento (stack) y de desapilamiento

El Z 80 posee instrucciones para apilar o desapilar registros de 16 bits, siendo el registro SP el puntero de pila (stack).

La operación de apilamiento se resume por las acciones siguientes:

SP ← SP - 1 :SP es decrementado en 1.  
 (SP) ← octeto de más peso :El octeto de más peso del registro es apilado (colocado en la dirección contenida en SP).

SP ← SP - 1 :SP es decrementado en 1.  
 (SP) ← octeto de menor peso :El octeto de menor peso del registro es apilado (colocado en la dirección contenida en SP).

La operación inversa de desapilamiento se resume:

Octeto de menos peso ← (SP) :El octeto de menos peso del registro es desapilado.

SP ← SP + 1 :SP es incrementado en 1.  
 Octeto de más peso ← (SP) :El octeto de más peso es desapilado.  
 SP ← SP + 1 :SP es incrementado en 1.

El cuadro siguiente da las instrucciones de apilamiento (PUSH) y de desapilamiento (POP) del Z 80.

Mnemotécnica	Operación	Código de la instrucción	Número de ciclos	Indicadores
PUSH qq	Apilamiento de qq	1 1 ←qq→ 0 1 0 1	11	No se modifican
PUSH IX	Apilamiento de IX	1 1 0 1 1 1 0 1 DD 1 1 1 0 0 1 0 1 E5	15	
PUSH IY	Apilamiento de IY	1 1 1 1 1 1 0 1 FD 1 1 1 0 0 1 0 1 E5	15	
POP qq	Desapilamiento de qq	1 1 ←qq→ 0 0 0 1	10	
POP IX	Desapilamiento de IX	1 1 0 1 1 1 0 1 DD 1 1 1 0 0 0 0 1 E1	14	
POP IY	Desapilamiento de IY	1 1 1 1 1 1 0 1 FD 1 1 1 0 0 0 0 1 E1	14	

Los valores posibles del registro doble qq son los siguientes:

Registro	qq
BC	00
DE	01
HL	10
AF	11

Estas instrucciones son útiles para guardar temporalmente el valor de un registro de 16 bits (PUSH) y restaurar este valor (POP) cuando sea necesario.

Cuando se utilicen estas instrucciones será necesario asegurarse de que el stack pointer contiene la dirección de una zona de memoria no utilizada. Será interesante poner una instrucción de inicialización del registro SP (LD SP, dirección zona libre) en cabeza de su programa.

Dado que los apilamientos se hacen decrementando el puntero del stack, el registro SP deberá ser iniciado a la dirección más alta de la zona libre (fondo de la pila).

### 1.6.10. Instrucciones de bifurcación

El Z 80 posee instrucciones de bifurcación incondicional que permiten saltar a una dirección de memoria sea cual fuere el estado de los indicadores (instrucciones análogas al GOTO del Basic). Igualmente tiene instrucciones de bifurcación condicional que sólo efectúan el salto si uno de los indicadores se encuentra en el estado deseado (es el equivalente al IF indicador = estado deseado THEN GOTO). Estas últimas instrucciones por lo general se colocan detrás de una instrucción de comprobación como CP.

Ejemplo:

CP 2 :A ¿es igual a 2?  
 JP Z,3E8H ;si es sí saltar a 3E8H, si no pasar a la instrucción siguiente.

La dirección de bifurcación puede ser definida de manera absoluta colocándola en dos octetos de la instrucción de salto (instrucción JP), o bien, de manera relativa, dándole el desplazamiento comprendido entre - 128 y + 127 que separa la dirección de bifurcación de la

dirección actual (instrucción JR) o también de manera indirecta mediante los registros HL, IX o IY (instrucción JP).

En la práctica, siempre que sea posible, tendremos interés en utilizar una instrucción de bifurcación relativa que sólo ocupa dos octetos, mejor que no una instrucción de bifurcación absoluta que ocupa tres octetos. Esta última se utilizará si la dirección de bifurcación se encuentra a más de 128 octetos de la dirección actual.

Mnemotécnico	Operación	Código de la instrucción	Número de ciclos	Indicadores
JP nn	Bifurcación absoluta a la dirección nn. PC ← nn	1 1 0 0 0 0 1 1 C3 ←n: menos peso → ←n: más peso →	10	No se modifican
JP cc, nn	Bifurcación condicional absoluta a la dirección nn. Si condición cc: PC ← nn	1 1 ←cc→ 0 1 0 ←n: menos peso → ←n: más peso →	10	
JR d	Bifurcación relativa de d octetos a partir de la instrucción siguiente a JR. PC ← PC + d	0 0 0 1 1 0 0 0 18 ← d →	12	
JR C, d	Si C = 1 PC ← PC + d Si no continuar	0 0 1 1 1 0 0 0 38 ← d →	12 si C = 1 7 si C = 0	
JR NC, d	Si C = 0 PC ← PC + d Si no continuar	0 0 1 1 0 0 0 0 30 ← d →	12 si C = 0 7 si C = 1	
JR Z, d	Si Z = 1 PC ← PC + d Si no continuar	0 0 1 0 1 0 0 0 28 ← d →	12 si Z = 1 7 si Z = 0	
JR NZ, d	Si Z = 0 PC ← PC + d Si no continuar	0 0 1 0 0 0 0 0 20 ← d →	12 si Z = 0 7 si Z = 1	
JP (HL)	Bifurcación indirecta HL: PC ← HL	1 1 1 0 1 0 0 1 E9	4	
JP (IX)	Bifurcación indirecta IX: PC ← IX	1 1 0 1 1 1 0 1 DD 1 1 1 0 1 0 0 1 E9	8	
JP (IY)	Bifurcación indirecta IY: PC ← IY	1 1 1 1 1 1 0 1 FD 1 1 1 0 1 0 0 1 E9	8	
DJNZ d	B ← B - 1 Decrementar B si B ≠ 0 PC ← PC + d Si no continuar	0 0 0 1 0 0 0 0 10 ← d →	13 si b ≠ 0 8 si b = 0	

La instrucción DJNZ es una instrucción de alto nivel que permite fácilmente programar bucles de un programa y simular así la instrucción FOR... NEXT del Basic. El registro B sirve entonces de contador de bucle. Debe ser inicializado a la cantidad de reiteraciones deseadas (número comprendido entre 0 y 255, el número 0 representa 256 reiteraciones).

La instrucción DJNZ tiene la función de la instrucción NEXT B cuando el paso es igual a - 1. Esta decremента el registro B y salta al principio del bucle si B es distinto de cero. Si no termina el bucle, pasando a la instrucción siguiente.

```

300 LD B,n ;inicialización del contador de bucle.
302 .
.
. cuerpo del bucle
.
.
312 DJNZ $ - 10H; retorno al principio del bucle si B ≠ 0
      (dirección 302)
dirección mnemotécnica
  
```

El símbolo cc de la tabla precedente designa una de las condiciones siguientes:

Condición	cc
NZ: no es cero	Z = 0 0 0 0
Z : cero	Z = 1 0 0 1
NC: no hay Carry	C = 0 0 1 0
C : Carry	C = 1 0 1 1
PO: paridad impar	P/V = 0 1 0 0
PE: paridad par	P/V = 1 1 0 1
P : signo positivo	S = 0 1 1 0
M : signo negativo	S = 1 1 1 1

### 1.6.11. Subprogramas

El Z 80 dispone de una instrucción de llamada a subprograma CALL y de una instrucción de retorno de subprograma RET que tienen respectivamente el papel de las instrucciones GOSUB y RETURN del Basic.



La instrucción CALL se resume por las operaciones siguientes:

SP ← SP - 1	} Puesta en el stack de la dirección de retorno que está contenida en PC.
(SP) ← octeto de más peso de PC	
SP ← SP - 1	
(SP) ← octeto de menos peso de PC	} Bifurcación a la dirección del subprograma (nn).
PC ← nn	

La instrucción RET se resume por las siguientes operaciones:

Octeto de menos peso de PC ← (SP)	} Desapilamiento en la dirección de retorno que se coloca en el registro PC.
SP ← SP + 1	
Octeto de más peso de PC ← (SP)	
SP ← SP + 1	

Mnemotécnico	Operación	Código de la instrucción	Número de ciclos	Indicadores
CALL nn	Llamada del subprograma situado en la dirección nn.	1 1 0 0 1 1 0 1 CD ← n: menos peso → ← n: más peso →	17	No se modifican
CALL cc, nn	Si cc es verdad, llamada del subprograma situado en la dirección nn. Si no continuar.	1 1 ← cc → 1 0 0 ← n: menos peso → ← n: más peso →	17 si cc es verdad 10 si cc es falso	
RET	Retorno de subprograma.	1 1 0 0 1 0 0 1 C9	10	
RET cc	Si cc es verdad, retorno del subprograma. Si no continuar.	1 1 ← cc → 0 0 0	11 si cc es verdad 5 si cc es falso	
RST p	Llamada del subprograma situado en la dirección p.	1 1 ← t → 1 1 1	11	

El significado de cc es el mismo que en el párrafo precedente.

La instrucción RST p llama a uno de los ocho subprogramas que se dan en la tabla siguiente, que suministra igualmente la correspondencia con los tres bits del código de la instrucción.

Dirección p	t
00H	0 0 0
08H	0 0 1
10H	0 1 0
18H	0 1 1
20H	1 0 0
28H	1 0 1
30H	1 1 0
38H	1 1 1

Estos subprogramas se sitúan en la ROM y por lo tanto no pueden ser modificados por el usuario. Son utilizados por el Basic y se procurará no emplearlos, a menos que se tomen todas las precauciones inherentes a su empleo.

Gracias a la estructura de pila utilizada podrán imbricarse varias llamadas a subprogramas; el primer RET terminará el programa más interior.

Siempre será interesante guardar los registros (PUSH) utilizados en el subprograma al principio del mismo y restaurar estos registros (POP) justo antes de la instrucción de retorno (RET). Deberá tomarse la precaución de tener tantos PUSH como POP en un subprograma con el fin de evitar que el retorno del mismo conduzca a una dirección cuyo valor sea el de un registro guardado en el stack. Este problema se debe a que la misma pila que se utiliza para los subprogramas también se utiliza para guardar los registros. Esta es la causa de numerosos errores de los principiantes que por lo general provoca un bloqueo del ordenador. En este caso se está obligado a desconectar el ordenador y volverlo a conectar. Evidentemente, el programa en memoria queda borrado. Así pues, siempre será preferible el guardar un programa ensamblado antes de comenzar su ejecución.

Ejemplo:

```
PUSH AF
PUSH BC guarda los registros utilizados
PUSH DE
```

.....  
cuerpo del subprograma  
.....

```
POP DE
POP BC restauración de los registros en orden inverso
POP AF
RET retorno del subprograma
```

### 1.6.12. Instrucciones de Intercambios

Estas son instrucciones que intercambian el valor de los números de 16 bits. En la tabla siguiente la operación de intercambio se simboliza por:  $\leftrightarrow$ .

Mnemotécnico	Operación	Código de la instrucción	Número de ciclos	Indicadores
EX DE, HL	DE $\leftrightarrow$ HL	1 1 1 0 1 0 1 1 EB	4	No se modifican
EX AF, AF'	AF $\leftrightarrow$ A'F'	0 0 0 0 1 0 0 0 0B	4	
EXX	BC $\leftrightarrow$ B'C' DE $\leftrightarrow$ D'E' HL $\leftrightarrow$ H'L'	1 1 0 1 1 0 0 1 D9	4	
EX (SP), HL	H $\leftrightarrow$ (SP + 1) L $\leftrightarrow$ (SP)	1 1 1 0 0 0 1 1 E3	19	
EX (SP), IX	Octeto más peso de IX $\leftrightarrow$ (SP + 1)	1 1 0 1 1 1 0 1 DD	23	
	Octeto menos peso de IX $\leftrightarrow$ (SP)	1 1 1 0 0 0 1 1 E3		
EX (SP), IY	Octeto más peso de IY $\leftrightarrow$ (SP + 1)	1 1 1 1 1 1 0 1 FD	23	
	Octeto menos peso de IY $\leftrightarrow$ (SP)	1 1 1 0 0 0 1 1 E3		

Estas instrucciones son de uso menos importante que las precedentes, pero en ciertos casos mejoran las cualidades o posibilidades de un programa en lenguaje máquina.

### 1.6.13. Instrucciones de transferencia de memoria y de búsqueda

En este apartado abordaremos las instrucciones de más alto nivel del Z 80 que tanto han contribuido a su éxito. Las instrucciones LDI, LDIR, LDD, LDDR, permiten hacer transferencias de octetos a memoria mientras que las instrucciones CPI, CPIR, CPD y CPDR sirven para buscar un octeto en la memoria.

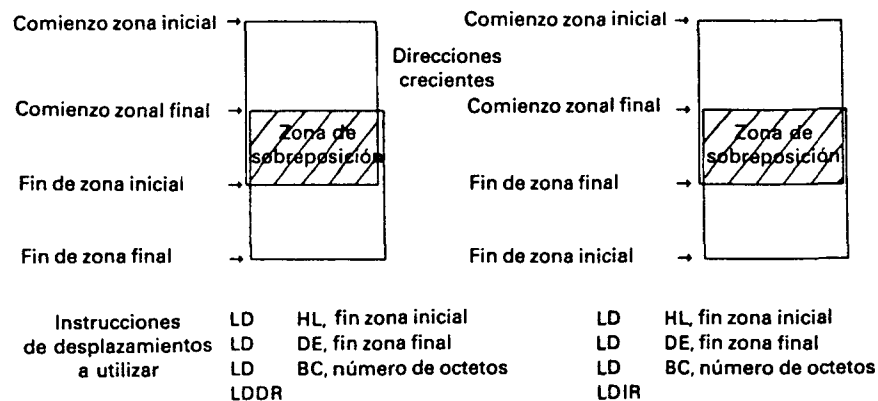
Mnemotécnico	Operación	Código de la instrucción	Número de ciclos	Indicadores
LDI	(DE) $\leftarrow$ (HL) DE $\leftarrow$ DE + 1 HL $\leftarrow$ HL + 1 BC $\leftarrow$ BC - 1	1 1 1 0 1 1 0 1 ED 1 0 1 0 0 0 0 0 A0	16	S, Z, C, no se modifican.
LDIR	(DE) $\leftarrow$ (HL) DE $\leftarrow$ DE + 1 HL $\leftarrow$ HL + 1 BC $\leftarrow$ BC - 1 Repetir mientras BC $\neq$ 0	1 1 1 0 1 1 0 1 ED 1 0 1 1 0 0 0 0 B0	16 + 21 * (BC - 1)	H y N puestos a cero. P/V puesto a cero para LDIR y LDDR.
LDD	(DE) $\leftarrow$ (HL) DE $\leftarrow$ DE - 1 HL $\leftarrow$ HL - 1 BC $\leftarrow$ BC - 1	1 1 1 0 1 1 0 1 ED 1 0 1 0 1 0 0 0 A8	16	Si no P/V vale 0 si BC = 1. (BC = 0 al final de la instrucción) y si no 1.
LDDR	(DE) $\leftarrow$ (HL) DE $\leftarrow$ DE - 1 HL $\leftarrow$ HL - 1 BC $\leftarrow$ BC - 1 Repetir mientras BC $\neq$ 0	1 1 1 0 1 1 0 1 ED 1 0 1 1 1 0 0 0 B8	16 + 21 * (BC - 1)	

La instrucción LDI transfiere el octeto situado en la dirección HL a la posición de memoria situada en la dirección DE. DE y HL son incrementados mientras que BC es decrementado.

La instrucción LDIR ejecuta la acción de la instrucción LDI en tanto que BC sea distinto de cero. BC desarrolla así la función de contador de bucle. Inicializando BC a la cantidad de octetos que se desean desplazar, HL a la dirección inicial de la zona a desplazar, DE con la dirección inicial de la zona destino y utilizando la instrucción LDIR, el número de octetos deseado se desplaza de la zona inicial hacia la zona final. La duración de esta instrucción es proporcional al número de octetos desplazados.

Las instrucciones LDD y LDDR son análogas a las instrucciones LDI y LDIR, pero realizan la transferencia en el sentido de direcciones decrecientes.

Según que la dirección inicial sea inferior o no a la dirección final, deberá utilizarse la instrucción LDDR en lugar de la instrucción LDIR si las zonas se recubren, para evitar que las zonas de recubrimiento sean mal desplazadas.



Examinemos ahora las instrucciones de búsqueda:

Mnemotécnico	Operación	Código de la instrucción	Número de ciclos	Indicadores
CPI	Comp. si A = (HL) ? HL ← HL + 1 BC ← BC - 1	1 1 1 0 1 1 0 1 ED 1 0 1 0 0 0 0 1 A1	16	S contiene el signo de A - (HL)
CPIR	Comp. si A = (HL) ? HL ← HL + 1 BC ← BC - 1 Repetir mientras A ≠ (HL) et BC ≠ 0	1 1 1 0 1 1 0 1 ED 1 0 1 1 0 0 0 1 B1	16 + 21 • (BC - 1)	H = 1 si hay un acarreo del bit 4 durante la comparación.
CPD	Comp. si A = (HL) ? HL ← HL - 1 BC ← BC - 1	1 1 1 0 1 1 0 1 ED 1 0 1 0 1 0 0 1 A9	16	P/V vale 0 si BC = 1 si no P/V vale 1.
CPDR	Comp. si A = (HL) ? HL ← HL - 1 BC ← BC - 1 Repetir mientras A ≠ (HL) et BC ≠ 0	1 1 1 0 1 1 0 1 ED 1 0 1 1 1 0 0 1 B9	16 + 21 • (BC - 1)	N vale 1. C no se modifica

Estas instrucciones buscan un octeto cuyo valor es idéntico al de A, a partir de la dirección HL en el sentido de las direcciones crecientes (CPI y CPIR), o en sentido inverso (CPD y CPDR) repitiendo la búsqueda, mientras que BC ≠ 0 (CPIR y CPDR) o sin repetición (CPI y CPD).

Estos dos grupos de instrucciones facilitan la programación de dos problemas muy corrientes: la transferencia a memoria y la búsqueda. Estas instrucciones son, aproximadamente, dos veces más rápidas que la serie de instrucciones elementales a las que representan. Será interesante, pues, utilizarlas lo más a menudo posible para que el código generado se más eficaz.

#### 1.6.14. Instrucciones relacionadas con las interrupciones

Una interrupción es una señal enviada a la entrada de interrupción del microprocesador que interrumpe a éste (y por lo tanto al programa que ejecutaba) para que efectúe una serie de instrucciones llamadas subprogramas de interrupción. En el ZX SPECTRUM, un reloj interno provoca una interrupción enmascarable por el microprocesador cada 200 milisegundos (500 interrupciones por segundo). El subprograma de interrupción situado en la dirección 38H hace una lectura del teclado e incrementa el reloj de tiempo real, utilizado entre otras por la instrucción PAUSE. Guarda todos los registros para no perturbar al programa interrumpido. Este último no es sensible a las interrupciones más que en el hecho de que tarda más tiempo en ejecutarse cuando es interrumpido. No obstante, el subprograma de interrupción utiliza el registro IY que contiene el valor 5C3AH. Por lo tanto, un programa en lenguaje máquina que permite las interrupciones (instrucción EI) no deberá emplear el registro IY.

Existen dos tipos de interrupción: las interrupciones no enmascarables ligadas a la entrada NMI del Z 80, y las interrupciones enmascarables ligadas a la entrada INT del Z 80. Estas últimas pueden ser autorizadas o prohibidas por el programa gracias a las instrucciones EI y DI.

El Z 80 tiene tres modos de interrupción llamados 0, 1 y 2.

En el modo 0, cuando hay una interrupción enmascarable el microprocesador ejecuta una de las ocho instrucciones RST, cuya dirección se presenta en el bus de datos en ese momento.

En el modo 1, que es en el que funciona el ZX SPECTRUM, cuando hay una interrupción el Z 80 ejecuta la instrucción RST 38H.

En el modo 2, que es el más sofisticado, la dirección del subprograma de interrupción se coloca en la posición de memoria situada en la dirección formada por el registro I (octeto de más peso) y el octeto presente en el bus de datos (octeto de menos peso). Este último octeto ha sido suministrado por el elemento que ha provocado la interrupción.

La tabla siguiente suministra todas las instrucciones del Z 80 relativas a las interrupciones.

Mnemotécnico	Operación	Código de la instrucción	Número de ciclos	Indicadores
DI	Prohibición de interrupciones.	1 1 1 1 0 0 1 1 F3	4	No se modifican
EI	Autorización de interrupciones.	1 1 1 1 1 0 1 1 FB	4	
IM 0	Paso a modo 0 de interrupción.	1 1 1 0 1 1 0 1 ED 0 1 0 0 0 1 1 0 46	8	
IM 1	Paso a modo 1 de interrupción.	1 1 1 0 1 1 0 1 ED 0 1 0 1 0 1 1 0 56	8	
IM 2	Paso a modo 2 de interrupción.	1 1 1 0 1 1 0 1 ED 0 1 0 1 1 1 1 0 5E	8	
RETI	Retorno de interrupción. Fin del subprograma de interrupción.	1 1 1 0 1 1 0 1 ED 0 1 0 0 1 1 0 1 4D	14	
RETN	Retorno de interrupción no enmascarable.	1 1 1 0 1 1 0 1 ED 0 1 0 0 0 1 0 1 45	14	

Quando se ejecuta un subprograma de interrupción, las interrupciones están prohibidas. El retorno al programa interrumpido mediante RETI o RSTN provoca la autorización de las interrupciones. Fuera de este detalle, el funcionamiento del subprograma de interrupción es análogo a un subprograma ordinario (apilamiento de la dirección de retorno en el momento de la llamada y desapilamiento de esta dirección cuando se hace el retorno).

*Nota:* Mediante la instrucción LD A,I o LD A,R es posible saber, en un momento dado, si las interrupciones están autorizadas. El indicador P/V está a 1 si las interrupciones están autorizadas y a 0 si no lo están.

### 1.6.15. Instrucciones de gestión de los ports de entrada/salida

El Z 80 posee 65536 ports de 8 bits destinados a gestionar a los periféricos. Estos ports pueden ser considerados como buses de 8 bits que realizan la conexión entre el microprocesador y los periféricos. Se numeran desde 0 a 65535.

Mnemotécnico	Operación	Código de la instrucción	Número de ciclos	Indicadores
IN A, (n)	Lectura en A del port de número An.	1 1 0 1 1 0 1 1 DB ← n →	11	No se modifican
IN r, (C)	Lectura en r del port de número (BC).	1 1 1 0 1 1 0 1 ED 0 1 ← r → 0 0 0	12	S = signo del dato. Z = 1 si el dato es nulo. P/V contiene la paridad. H y N puestos a 0. C no se modifica.
OUT (n), A	Escritura de A en el port de número An.	1 1 0 1 0 0 1 1 D3 ← n →	11	No se modifican
OUT (C), r	Escritura de r en el port de número (BC).	1 1 1 0 1 1 0 1 ED 0 1 ← r → 0 0 1	12	No se modifican
INI	(HL) ← <port BC> B ← B - 1 HL ← HL + 1	1 1 1 0 1 1 0 1 ED 1 0 1 0 0 0 1 0 A2	16	S, H, P/V desconocidos Z = 1 si B = 1 al comienzo de la instrucción. N puesto a 1. C no es modificado.
IND	(HL) ← <port BC> B ← B - 1 HL ← HL - 1	1 1 1 0 1 1 0 1 ED 1 0 1 0 1 0 1 0 AA	16	S, H, P/V desconocidos Z y N son puestos a 1
INIR	(HL) ← <port BC> B ← B - 1 HL ← HL + 1 Repetir mientras B ≠ 0	1 1 1 0 1 1 0 1 ED 1 0 1 1 0 0 1 0 B2	16 + 21 • (B - 1)	S, H, P/V desconocidos Z y N son puestos a 1
INDR	(HL) ← <port BC> B ← B - 1 HL ← HL - 1 Repetir mientras B ≠ 0	1 1 1 0 1 1 0 1 ED 1 0 1 1 1 0 1 0 BA	16 + 21 • (B - 1)	C no es modificado
OUTI	<port BC> ← (HL) B ← B - 1 HL ← HL + 1	1 1 1 0 1 1 0 1 ED 1 0 1 0 0 0 1 1 A3	16	S, H, P/V desconocidos Z = 1 si B = 1 al comienzo de la instrucción. N puesto a 1. C no es modificado.
OUTD	<port BC> ← (HL) B ← B - 1 HL ← HL - 1	1 1 1 0 1 1 0 1 ED 1 0 1 0 1 0 1 1 AB	16	S, H, P/V desconocidos Z y N son puestos a 1
OTIR	<port BC> ← (HL) B ← B - 1 HL ← HL + 1 Repetir mientras B ≠ 0	1 1 1 0 1 1 0 1 ED 1 0 1 1 0 0 1 1 B3	16 + 21 • (B - 1)	S, H, P/V desconocidos Z y N son puestos a 1
OTDR	<port BC> ← (HL) B ← B - 1 HL ← HL - 1 Repetir mientras B ≠ 0	1 1 1 0 1 1 0 1 ED 1 0 1 1 1 0 1 1 BB	16 + 21 • (B - 1)	C no es modificado

Como veremos en el capítulo 4, el ZX SPECTRUM utiliza el port 254 para la gestión del altavoz, del lector de cassette y del borde de la pantalla; el port 251 para la gestión de la impresora, así como otros 8 ports para la gestión del teclado.

La lectura de un port se realiza mediante la instrucción IN que transfiere su contenido a un registro de 8 bits. La escritura del port se hace mediante la instrucción OUT que transfiere el contenido de un registro hacia el port.

El Z 80 tiene instrucciones de lectura y de escritura múltiples que efectúan una transferencia de datos entre el port y una zona de la memoria. El registro HL sirve de puntero de memoria y el registro B de contador del número de octetos transferidos. Estas instrucciones de alto nivel son de un relativo poco interés en el SPECTRUM.

En todas las instrucciones de entrada/salida del Z 80 el número del port está en el registro BC, o en el A (octeto de más peso) y en un entero n (octeto de menor peso).

## 1.7. Instrucciones Basic relativas a la utilización del lenguaje máquina

El Basic del ZX SPECTRUM tiene seis instrucciones que permiten utilizar programas en lenguaje máquina.

La instrucción *POKE dirección, octeto* sirve para modificar un octeto de la memoria. Equivale a las dos instrucciones de ensamblador siguientes:

LD A, octeto  
LD (dirección), A

La función *PEEK dirección* permite leer un octeto de la memoria. Es equivalente a:

LD A, (dirección)

El valor leído podrá ser visualizado mediante: *PRINT PEEK dirección*.

La función *USR dirección* provoca la bifurcación del Z 80 a la dirección especificada. Es equivalente a *CALL dirección*. Dado que *USR* es una función, deberá ser utilizada con otra instrucción como *PRINT*. Así la orden *PRINT USR dirección* provocará la bifurcación especificada. Si el puntero del stack no es modificado por el programa en lenguaje máquina colocado en esta dirección, la instrucción *RET* producirá el retorno al Basic.

La instrucción *CLEAR dirección* fija la dirección más alta utilizable por un programa Basic. Los octetos situados más allá de esta direc-

ción podrán emplearse por un programa en lenguaje máquina sin riesgo de que el programa Basic modifique estos octetos. Cuando se utilice a la vez un programa Basic y un programa en lenguaje máquina será necesario efectuar esta instrucción para evitar los conflictos que seguramente se producirán entre los dos programas.

La instrucción *SAVE «nombre» CODE dirección inicial, longitud* transfiere la zona de memoria que empieza en la dirección especificada a un fichero en cassette con el nombre especificado. Esta instrucción permite guardar los programas en lenguaje máquina.

La instrucción *LOAD «nombre» CODE* permitirá realizar la transferencia inversa para cargar en memoria el fichero cuyo nombre se especifica, el cual contiene un programa en lenguaje máquina.

## 2. Útiles de programación en ensamblador

En este capítulo describiremos cómo utilizar el Editor/Ensamblador y el Debugger sobre un ejemplo de programa.

Existen actualmente varios Ensambladores y varios Debuggers para el ZX SPECTRUM, pero esencialmente sólo describiremos el Editor/Ensamblador y el Debugger. No obstante, las ideas dadas en este capítulo serán válidas para otros útiles de desarrollo en Ensamblador. Solamente varía la sintaxis. Así pues, insistiremos más sobre el fin de las órdenes que sobre su sintaxis que ya se explica en los manuales de usuario.

### 2.1. Utilización del Editor/Ensamblador

Examinemos un programa capaz de transferir un número dado de octetos de un lugar de la memoria a otro. Este programa puede ser simplemente escrito mediante la utilización de la instrucción LDIR. Supongamos que queremos transferir 10 octetos situados a partir de la dirección 7900H hacia el espacio memoria que empieza en 7A00H. El programa se escribirá en mnemónicos:

```
LD HL, 7900H
LD DE, 7A00H
LD BC, 10
LDIR
```

Carguemos el Editor/Ensamblador y tecleemos el programa. El texto siguiente se visualizará en la pantalla:

			Comentarios
ZX SPECTRUM EDITOR/ENSAMBLADOR			Mensaje del autor visualizado en la introducción
Autor © 1983 Pascal PELLIER y Ediciones EYROLES			
> I			Orden de inserción de texto
0010	LD	HL,7900H	
0020	LD	DE,7A00H	Escritura del programa con utilización de caracteres de tabulación <TAB>
0030	LD	BC,10	
0040	LDIR		
0050			Pulsar <BREAK>
> A			Orden de ensamblaje
0000	210079	0010	LD
	HL,7900H		
0003	11007A	0020	LD
	DE,7A00H		
0006	010A00	0030	LD
	BC,10		
0009	EDB0	0040	LDIR
No hay END			
00001 Error (s)			
00000 es la dirección de lanzamiento			
			Visualización del texto ensamblado
			Visualización de un error

Después del ensamblaje, el Editor/Ensamblador visualiza para cada una de las instrucciones la dirección donde está situada, el código de la instrucción con 1, 2, 3 o 4 octetos, el número de línea correspondiente al mnemónico de la instrucción y los operandos. Así, la instrucción LD DE, 7A00H es colocada en la dirección 0003 (hexadecimal) y su código es 11007A (3 octetos).

Tal como está, el programa se colocará en la dirección cero, es decir, en memoria muerta (ROM). No podría ser ejecutado, puesto que no se le puede colocar en memoria muerta. Para colocarlo en otro lugar hay que indicar al ensamblador la dirección donde empieza con una orden ORG (origen). Una orden de ensamblaje es una pseudoinstrucción que permite actuar sobre el desarrollo del ensamblaje. Se coloca en una línea del programa en lenguaje ensamblador de la misma forma que una instrucción del Z 80.

Para colocar nuestro programa en la dirección 7800H (memoria viva), deberá añadirse en cabeza del programa la orden: ORG 7800H.

```
>I5
0006          ORG      7800H      Inserción en línea 5
No queda espacio entre líneas
```

Cuando se ejecuta la orden de inserción el Editor intenta insertar una línea después de la última línea insertada. Para esto añade el paso de inserción (10 por defecto) al último número de la línea insertada. Si este número es superior al número de la próxima línea del texto no podrá realizarse la inserción y visualizará: «No hay más lugar entre

las líneas». Esto es lo que se ha producido debido a la inserción de la línea 5.

En el programa que hemos teclado, el ensamblador señala el error «Sin END». Todo programa ensamblador debe terminar normalmente con la orden END, que indica el fin del programa y que especifica su dirección de desplazamiento, es decir, la dirección donde la ejecución debe empezar. Si nuestro programa está colocado en la 7800H, la dirección de lanzamiento será 7800H. Deberemos, pues, colocar la directiva END 7800H.

Si deseamos cambiar la dirección donde esté situado será necesario cambiar a la vez la orden ORG y la orden END. Es posible evitar esto empleando una etiqueta simbólica. Una etiqueta es una serie de caracteres alfanuméricos que representan una dirección de memoria. La definición de una etiqueta se hace colocándola justo después del número de línea en una instrucción. Seguidamente podremos hacer referencia al valor de la etiqueta colocándola en cualquier expresión aritmética autorizada por el ensamblador.

En este ejemplo coloquemos una etiqueta INICIO en la primera instrucción. Esta etiqueta representa la dirección de lanzamiento del programa. Por lo tanto podrá ser utilizada con la directiva END.

>E10			
0010 INICIO	LD	HL,7900H	Modificación de la línea 10 para la etiqueta INICIO
>I\$			
0050	END	INICIO	Inserción de la directiva END al final del programa
0060			
>N,10			
>P#,\$			Renumeración del texto
0010	ORG	7800H	
0020 INICIO	LD	HL,7900H	Visualización del texto
0030	LD	DE,7A00H	
0040	LD	BC,10	
0050	LDIF		
0060	END	INICIO	

El texto que acabamos de grabar se llama versión *fuentes* del programa. Es posible guardarlo en cassette a través de la orden SAVE del editor.

> S TRANSFERT	Guarda el programa con el nombre:
Preparar el cassette	«TRANSFERT».

La versión en lenguaje máquina del programa o versión *objeto* puede guardarse en cassette gracias a la opción WO (Write Output) del ensamblador:

>AOBJET-WO			
7800	0010	ORG	Ensamblaje del programa y creación del objeto sobre cassette con el nombre: «OBJET»
7800H			
7800 210079	0020 INICIO	LD	
HL,7900H			
7803 11007A	0030	LD	
DE,7A00H			
7806 010A00	0040	LD	
BC,10			
7809 EDB0	0050	LDIF	
7800	0060	END	
INICIO			
00000 Error (s)			
30720 es la dirección de lanzamiento			Dirección de lanzamiento 30720 = 7800H
Preparar el cassette			
>			

Utilicemos ahora este programa para transferir de un lugar a otro un mensaje constituido por caracteres ASCII. El código ASCII permite presentar todos los caracteres alfanuméricos en los 7 bits menos significativos de un octeto. En el SPECTRUM, el octavo bit está a cero. Existe una correspondencia biunívoca entre un carácter y su código ASCII. La orden DEFM del ensamblador asegura la transcripción en códigos ASCII de una serie de caracteres dados entre comillas y genera los caracteres correspondientes.

Ejemplo:

DEFM «Bonjour»

Esta orden genera sucesivamente los 7 octetos que representan a los códigos ASCII de cada una de las letras del nombre «Bonjour».

Empleemos ahora la orden EQU para dar a la etiqueta LONG la longitud del mensaje asociado a la directiva DEFM.

FUENTE	DEFM	«Bonjour»
LONG	EQU	\$—FUENTE

La expresión \$—FUENTE es igual a la longitud del mensaje, puesto que \$ designa la dirección actual; es decir, la dirección del último octeto del mensaje más 1 y FUENTE es la dirección del primer octeto del mensaje. Este valor es colocado en la etiqueta LONG por la orden EQU. El uso de esta etiqueta hará el programa independiente de la longitud del mensaje utilizado.

Para especificar el lugar donde vamos a transferir el mensaje emplearemos la orden DEFS que permite reservar cierto número de octetos para guardar variables o tablas:

DEST            DEFS            LONG

Esta orden reserva un número igual a LONG de octetos. DEST contiene la dirección de la zona de memoria reservada.

Efectuemos estas modificaciones en el programa y examinemos el código generado.

```
>E 20
0020 INICIO    LD    HL,FUENTE    Modificación de la línea 20.
0030           LD    DE,7A00H    Pasar a la línea siguiente (↓)
>E
0030           LD    DE,DEST
0040           LD    BC,10       Modificación de la línea en curso
>E
0040           LD    BC,LONG
>I51,1
0051 FUENTE    DEFM 'Bonjour'    Inserción de tres directivas
0052 LONG      EQU    $-FUENTE    suplementarias
0053 DEST      DEFS    LONG
0054
>N,,10
>A-WS
7800           0010        ORG        Renumeración.
              7800H                    Ensamblaje con visualización
7800 210B78    0020 INICIO    LD        de símbolos
              HL,FUENTE
7803 111278    0030            LD
              DE,DEST
7806 010700    0040            LD
              BC,LONG
7809 ED80      0050            LDIR
780B 42        0060 FUENTE    DFEM
              'Bonjour'
              6F 6E 6A 6F 75 72
0007           0070 LONG    EQU        Códigos ASCII de «Bonjour».
              $-FUENTE                Longitud de Bonjour = 7
7812           0080 DEST    DEFS
              LONG
7800           0090            END
              INICIO
00000 Error (s)
30720 es la dirección de lanzamiento

INICIO    7800    DEST    7812
LONG      0007    FUENTE    780B
>
```

Para terminar, tomaremos de nuevo nuestro programa y lo modificaremos de manera que prescindamos de la instrucción LDIR mediante una serie de 8 instrucciones que formen un bucle. La última instruc-

ción es una instrucción de salto que permite el retorno al principio del bucle simbolizado por la etiqueta LOOP.

```
0010 INICIO    ORG    7800H
0020           LD    HL,FUENTE
0030           LD    DE,DEST
0040           LD    BC,LONG
0080 LOOP      LD    A,(HL)        ; lectura de un carácter
0060           LD    (DE),A        ; transferencia del carácter
0070           INC    HL            ; aumentar los punteros
0080           INC    DE
0090           DEC    BC            ; decrementar el contador de bucle
0100           LD    A,B            ; comprueba si BC es nulo
0110           OR    C
0120           JR    NZ,LOOP       ; bifurcación si la transferencia
0130 FUENTE    DEFM 'Bonjour'    ; no ha terminado
0140 LONG      EQU    $-FUENTE
0160 DEST      DEFS    LONG
0160           END    INICIO
```

En estos ejemplos nos hemos percatado de algunas de las posibilidades de edición del Editor/Ensamblador. Posee muchas más que serán de utilidad cuando se escriban programas más importantes.

Igualmente hemos visto el interés de utilizar etiquetas simbólicas y órdenes de ensamblaje. Las etiquetas permiten especificar simplemente una posición. El texto fuente podrá fácilmente ser trasladado de un lugar a otro de la memoria, mediante la orden ORG, encargándose el ensamblador de recalculer todas las direcciones de las etiquetas. Las órdenes de ensamblaje permiten modificar el desarrollo del ensamblaje (ORG, END) o solicitan la generación de constantes o espacios para las variables (DEFM, DEFB, DEFW, DEFS). La lista completa y detallada de estas órdenes se encuentra en el manual que acompaña al Editor/Ensamblador.



## 2.2. Utilización del Debugger

Carguemos el Debugger y la versión objeto de nuestro último programa de transferencia utilizando la orden L (LOAD) del Debugger. Esta visualiza el nombre del programa cargado y su situación en memoria (principio-fin).

```
ZX SPECTRUM DEBUGGER 6CB6 - 74D2
Autor © 1983 Pascal PELLIER
y Ediciones EYROLES
```

Mensaje del autor

```
0000 F3          DI
>L
Programa      OBJET          7800-7818
      ↑
      Nombre del programa      Situación en memoria
```

Instrucción en curso (PC = 0)  
Orden de carga

El programa OBJETO está cargado en memoria; desensamblémoslo en mnemotécnicos del Z 80.

```
>D7800H,7811H
7800 21 12 78    LD    HL,7812
7803 11 19 78    LD    DE,7819
7806 01 07 00    LD    BC,0007
7809 7E          LD    A,(HL)
780A 12          LD    (DE),A
780B 23          INC   HL
780C 13          INC   DE
780D 0B          DEC   BC
780E 78          LD    A,B
780F B1          OR    C
7810 20 F7      JR    NZ,7809
Direc. Código      Mnemotécnico
```

Orden de desensamblaje entre las direcciones 7800H y 7811H

Nos encontramos con el texto fuente de nuestro programa, donde las etiquetas que son propias al Editor/Ensamblador han sido reemplazadas por su valor en hexadecimal.

Visualicemos esta parte de la memoria en hexadecimal y en ASCII para examinar las variables y los datos:

```
> M 7800H
```

ADDRESSES	7800	A 787F	ASCII			
00	2112	7811	1978	0107	l . x . . x . .	
08	007E	1223	130B	78B1	. " . # . . . x .	
10	20F7	426F	6E6A	6F78	. Bonjour	Mensaje «Bonjour»
18	7200	0000	0000	0000	.....	
20	0000	0000	0000	0000	.....	
28	0000	0000	0000	0000	.....	
30	0000	0000	0000	0000	.....	
38	0000	0000	0000	0000	.....	
40	0000	0000	0000	0000	.....	
48	0000	0000	0000	0000	.....	
50	0000	0000	0000	0000	.....	
58	0000	0000	0000	0000	.....	
60	0000	0000	0000	0000	.....	
68	0000	0000	0000	0000	.....	
70	0000	0000	0000	0000	.....	
78	0000	0000	0000	0000	.....	

Ejecutemos ahora el programa paso a paso (instrucción tras instrucción) gracias a la orden @ del Debugger. Pero antes inicialicemos el contador ordinal (registro PC) al valor 7800H que es la dirección de lanzamiento de nuestro programa.

```
>RPC = 7800H
```

```
AF=0000 BC=0000 DE=0000 HL=0000   Registros primarios
AF=0000 BC=0000 DE=0000 HL=0000   Registros secundarios
IX=0000 IY=6C3A SP=FFFF PC=7800   Registro de 16 bits
```

Esta orden visualiza el valor de todos los registros, la mayoría de los cuales están inicializados a cero. Los registros de 8 bits se agrupan para formar registros dobles.

Ejecutemos paso a paso nuestro programa pulsando @ para cada instrucción y examinemos el valor de los registros en el curso de la ejecución.

```

7800 21 12 78 LD HL,7812
AF=0000 BC=0000 DE=0000 HL=7812
'AF=0000 BC=0000 DE=0000 HL=0000
IX=0000 IY=5C3A SP=FFFF PC=7803
7803 11 19 78 LD DE,7819
AF=0000 BC=0000 DE=7618 HL=7612
'AF=0000 BC=0000 DE=0000 HL=0000
IX=0000 IY=5C3A SP=FFFF PC=7806
7806 01 07 00 LD BC,0007
AF=0000 BC=0007 DE=7619 HL=7812
'AF=0000 BC=0000 DE=0000 HL=0000
IX=0000 IY=5C3A SP=FFFF PC=7809
7809 7E LD A,(HL)
AF=4200 BC=0007 DE=7819 HL=7812
'AF=0000 BC=0000 DE=0000 HL=0000
IX=0000 IY=5C3A SP=FFFF PC=780A
780A 12 LD (DE),A
AF=4200 BC=0007 DE=7819 HL=7812
'AF=0000 BC=0000 DE=0000 HL=0000
IX=0000 IY=5C3A SP=FFFF PC=780B
780B 23 INC HL
AF=4200 BC=0007 DE=7819 HL=7813
'AF=0000 BC=0000 DE=0000 HL=0000
IX=0000 IY=5C3A SP=FFFF PC=780C
780C 15 INC DE
AF=4200 BC=0007 DE=781A HL=7813
'AF=0000 BC=0000 DE=0000 HL=0000
IX=0000 IY=5C3A SP=FFFF PC=780D
780D 0B DEC BC
AF=4200 BC=0006 DE=781A HL=7813
'AF=0000 BC=0000 DE=0000 HL=0000
IX=0000 IY=5C3A SP=FFFF PC=780E
780E 78 LD A,B
AF=0000 BC=0005 DE=781A HL=7813
'AF=0000 BC=0000 DE=0000 HL=0000
IX=0000 IY=5C3A SP=FFFF PC=780F
780F B1 DR C
AF=0004 BC=0006 DE=781A HL=7813
'AF=0000 BC=0000 DE=0000 HL=0000
IX=0000 IY=5C3A SP=FFFF PC=7810
7810 20 F7 JR NZ,7809
AF=0604 BC=0006 DE=781A HL=7813
'AF=0000 BC=0000 DE=0000 HL=0000
IX=0000 IY=5C3A SP=FFFF PC=7809
7809 7E LD A,(HL)
AF=6F04 BC=0006 DE=781A HL=7813
'AF=0000 BC=0000 DE=0000 HL=0000
IX=0000 IY=5C3A SP=FFFF PC=780A
780A 12 LD (DE),A

```

```

Primera instrucción
HL es inicializado
DE es inicializado
BC es inicializado
Inicio del bucle
Transferencia de un octeto
Comprueba el fin del bucle
Segunda ejecución del bucle
Etc...

```

El programa se termina cuando el registro PC alcanza el valor 7812H. Entonces el valor de los registros es el siguiente:

```

AF=0044 BC=0000 DE=7820 HL=7819
'AF=0000 BC=0000 DE=0000 HL=0000
IX=0000 IY=5C3A DE=FFFF PC=7812

```

La visualización de la página de memoria que empieza en 7800H permite verificar que la transferencia ha sido bien hecha:

ADRESSES	7800	A 787F	ASCII		
00	2112	7811	1978	0107	l . x . . x . .
08	007E	1223	130B	78B1	. " . * . . x .
10	20F7	426F	6E6A	6F75	. Bonjour
18	7242	6F6E	6A6F	7572	. Bonjour
20	0000	0000	0000	0000	.....
28	0000	0000	0000	0000	.....
30	0000	0000	0000	0000	.....
38	0000	0000	0000	0000	.....
40	0000	0000	0000	0000	.....
48	0000	0000	0000	0000	.....
50	0000	0000	0000	0000	.....
58	0000	0000	0000	0000	.....
60	0000	0000	0000	0000	.....
68	0000	0000	0000	0000	.....
70	0000	0000	0000	0000	.....
78	0000	0000	0000	0000	.....

Duplicado del mensaje «Bonjour»

Es posible ejecutar el programa normalmente utilizando la orden G (GO) que permite lanzar la ejecución a partir de una dirección dada por el usuario. Si se efectúa la orden G 7800H directamente, el programa se ejecutará, ya que el microprocesador ejecutará todos los octetos que se encuentran después del programa en memoria (en principio éstos son octetos nulos: instrucción NOP). Cuando el registro PC haya conseguido el valor FFFFH, volverá a la dirección 0, lo que provocará una reinicialización del ZX SPECTRUM y un borrado de la memoria.

Para evitar este inconveniente hay que colocar un punto de paro al final del programa con la letra B (Break point). Un punto de paro, de hecho, es una instrucción de salto al Debugger, constituida por tres octetos que se colocan en la dirección especificada por el usuario. Esta instrucción se coloca en el momento de la orden G y se suprime después del retorno al Debugger, de manera que su uso sea invisible para el usuario.

En nuestro ejemplo coloquemos el punto de paro número cero (pueden utilizarse hasta 10 puntos de paro) en la dirección 7812H mediante la orden:

```

B0 = 7812H
↑
número de punto de paro

```

Ejecutemos de una sola vez el programa mediante la orden:

G7800H

Hay retorno al Debugger con la visualización del valor de los registros, que es la misma que anteriormente.

Examinemos la página de memoria situada en 7800H para ver si la transferencia es correcta:

ADDRESSES	7800	A 787F	ASCII		
00	2112	7811	1978	0107	l . x . . x . .
08	007E	1223	130B	78B1	. . # . . x .
10	20F7	426F	6E6A	6F75	. Bonjour
18	7200	0065	6A6F	7572	. . . sjour
20	0000	0000	0000	0000	.....
28	0000	0000	0000	0000	.....
30	0000	0000	0000	0000	.....
38	0000	0000	0000	0000	.....
40	0000	0000	0000	0000	.....
48	0000	0000	0000	0000	.....
50	0000	0000	0000	0000	.....
58	0000	0000	0000	0000	.....
60	0000	0000	0000	0000	.....
68	0000	0000	0000	0000	.....
70	0000	0000	0000	0000	.....
78	0000	0000	0000	0000	.....

Los cuatro últimos octetos del nombre «Bonjour» se han transferido correctamente, mientras que los tres primeros han tomado los valores hexadecimales CD, 0C y 65 que son de hecho los tres octetos de la instrucción de retorno al Debugger. Al estar el punto de paro colocado en la misma dirección que el mensaje «Bonjour» inicial, éstos son los valores del código de la instrucción de salto que se han utilizado para la transferencia de los tres primeros octetos. Al final de la ejecución esta instrucción de salto ha sido suprimida, lo que explica que ya no sea visible en el mensaje «Bonjour» inicial.

En la práctica habrá que tener en cuenta estos tres octetos después del empleo de los puntos de paro para evitar funcionamientos incorrectos o incluso «pérdidas» del programa.

Acabamos de ver en un ejemplo la utilización de las principales órdenes del Debugger. Posee otras, pero dejamos al lector que las experimente siguiendo el manual que se adjunta con el Debugger.

### 2.3. Utilización del Editor/Ensamblador con el Debugger

Con un SPECTRUM 48 K es posible utilizar el Debugger conjuntamente con el Editor/Ensamblador.

Para ello cargue el Debugger después de la conexión y colóquelo en la parte más alta de la memoria por medio de la orden PE 700H.

Vuelva al monitor Basic mediante G121CH.

Cargue el Editor/Ensamblador que se coloca en la parte baja de la memoria (dirección 5E53H).

Escriba su programa o modifique un programa fuente previamente guardado en cassette.

Ensamble este programa en memoria (A-IM) colocándolo en un lugar en el que no apiaste al Debugger y que se encuentre después del texto fuente de su programa (el ensamblador visualizará el mensaje «Dirección incorrecta» si no ocurre así). Utilice la orden U del ensamblador para conocer el tamaño de la zona libre. Reste a esta cantidad algunas centenas de octetos para la tabla de símbolos. Restando el resultado de FFFFH obtendrá la dirección mínima de colocación de su programa. Entonces deberá verificar, después del ensamblaje, que no haya errores de «Dirección incorrecta» y que la última dirección sea inferior a E700H.

Si se han satisfecho todas estas condiciones, podrá pasar al Debugger mediante la orden BE81EH. Será prudente guardar previamente su programa fuente en cassette.

Ejecute el programa paso a paso o con la orden GO colocando puntos de paro.

Para volver al Editor/Ensamblador, sin perder el programa fuente en memoria, ejecute la orden G5E61H cuando las interrupciones estén autorizadas (orden E del Debugger).

Así se puede realizar una especie de va y viene entre el Editor/Ensamblador y el Debugger hasta que el programa esté totalmente a punto.

Evidentemente, este procedimiento sólo es válido si su programa no es demasiado grande para que quepa todo en la memoria. No obstante, permite escribir programas harto consistentes. Será particularmente útil para aquellos que empiezan en la programación en lenguaje máquina.

### 3. Subprogramas de interés general

En este capítulo describiremos cierta cantidad de subprogramas que serán útiles para la mayor parte de los programas en ensamblador, tanto para juegos de acción rápida como para programas más serios.

Para cada subprograma daremos una explicación de su funcionamiento, su modo de empleo, así como el listado del ensamblaje abundantemente comentado. El subprograma siempre será colocado en la dirección cero por razones de homogeneidad. Si desea probarlo sobre su microordenador modifique su situación de manera que se cargue sobre la memoria viva (RAM) y que no aplaste al Debugger que le servirá para probarlo.

#### 3.1. Multiplicación de números enteros: MUL

Este programa emplea un procedimiento análogo al que utilizamos cuando efectuamos una multiplicación en binario a mano. Tomemos por ejemplo la multiplicación  $1010$  por  $1100$ ; obtenemos una operación de este tipo:

$1010$	multiplicando
$\times 1100$	multiplicador
$0000$	
$0000$	
$1010$	productos parciales
$1010$	
$111000$	resultado

Los cuatro productos parciales, en este ejemplo valen  $0000$ , o bien  $1010$ , según que el bit correspondiente del multiplicador valga  $0$  o  $1$ . El resultado se obtendrá realizando la suma de los diversos productos parciales convenientemente decalados. Únicamente hemos utilizado operaciones de comprobación, de sumas y de decalaje, que el microprocesador sabe efectuar. El algoritmo utilizado en el programa se deriva de esto que acabamos de decir, procediendo siempre en

sentido inverso por razones de comodidad. Se empieza por tratar los bits de más peso del multiplicador y se decala cada vez el resultado. Si el bit del multiplicador está a  $1$  se le añade el multiplicando; de lo contrario no se hace nada. La operación se termina cuando el bit de menos peso ha sido tratado.

```

00010 :.....
00020 :
00030 : SUBPROGRAMA DE MULTIPLICACION ENTERA DE UN
00040 : NUMERO DE 16 BITS POR UN NUMERO DE 8 BITS
00080 :
00090 : ENTRADA: DE CONTIENE EL MULTIPLICANDO
00070 : A CONTIENE EL MULTIPLICADOR
00080 :
00090 : SALIDA: HL CONTIENE EL RESULTADO DE LA MULTIPLICACION
00100 :
00110 :
00120 :.....
0000 210000 00130 LD HL,0 ; INICIALIZACION DEL RESULTADO
0003 0608 00140 LD B,B ; NUMERO DE BITS A MULTIPLICAR
0005 29 00150 HO ADD HL,HL ; EL RESULTADO ES MULTIPLICADO POR 2
0006 CB27 00160 SLA A ; EL BIT DE MAS PESO DE A ES
; PUESTO EN EL CARRY
0008 3001 00180 JR NC,H1 ; SI ESTE BIT VALE 0
000A 19 00190 ADD HL,DE ; SI NO AÑADIR EL MULTIPLICANDO
000B 10F8 00200 H1 DJNZ HO ; TRATAR TODOS LOS BITS
000D C9 00210 RET ; FIN DEL SUBPROGRAMA
0000 00220 END
00000 TOTAL ERRORES
    
```

El programa efectúa una multiplicación de un número de 16 bits por un número de 8 bits y da un resultado de 16 bits. Así, si los números son demasiado grandes, puede que el resultado no quepa en 16 bits, entonces hay un desbordamiento y el resultado es erróneo. Por consiguiente, habrá que asegurarse que no ocurra esto cuando se emplee esta rutina de multiplicación.

#### 3.2. División de números enteros: DIV

El algoritmo utilizado en estos subprogramas proviene del procedimiento empleado cuando se efectúa una división binaria a mano. Tomemos como ejemplo la división  $1011011$  por  $1110$ :

dividendo	$1011011$	$1110$	divisor
restos parciales	$10110$	$0110$	cociente
	$10001$		
resto	$00111$		

El cociente vale  $110$  y el resto  $111$ .

El procedimiento utilizado en el programa es el siguiente: el registro A destinado a recibir los restos parciales es inicializado a cero.

Se pone el bit de más peso del dividendo en el bit  $0$  de A. Se mira entonces si A es superior al divisor. Si no lo es, se coloca un bit  $0$  en el

cociente, si no se coloca un bit 1 y se sustrae el divisor de A. Seguidamente A es decalado hacia la izquierda y se reemprende el proceso hasta que todos los bits del dividendo sean tratados.

Ejemplo:

	Valor de A	Cociente	
ciclo 0	0	0	} A es inferior al divisor
ciclo 1	1	0	
ciclo 2	10	0	
ciclo 3	101	0	
ciclo 4	1011	0	} A es superior al divisor
ciclo 5	10110	1	
ciclo 6	10001	1	} A es inferior al divisor
ciclo 7	111	0	

El valor del cociente, leído de arriba abajo, es igual a 0000110 y el resto está contenido en A (111).

```

00010 .....
00020 .....
00030 : SUBPROGRAMA DE DIVISION ENTERA DE UN
00040 : NUMERO DE 16 BITS POR UN NUMERO DE 8 BITS
00050 .....
00060 : ENTRADA: HL CONTIENE EL DIVIDENDO
00070 : C CONTIENE EL DIVISOR
00080 .....
00090 : SALIDA: HL CONTIENE EL COCIENTE
00100 : A CONTIENE EL RESTO
00110 .....
00120 .....
00130 .....
0000 AF 00140 DIV XOR A ; A CONTENDRA LOS PRIMEROS BITS
00150 : DEL DIVIDENDO
0001 0610 00160 LD B,16 ; NUMERO DE BITS DEL DIVIDENDO
0003 29 00170 H2 ADD HL,HL ; EL BIT 16 DEL DIVIDENDO ES
00180 : COLOCADO EN EL CARRY
0004 17 00190 RLA ; ESTE BIT ES COLOCADO EL DE
00200 : MENOR PESO DE A
0005 3803 00210 JR C,H4 ; SI HAY DESBORDAMIENTO DE A, A ES
00220 : SUPERIOR AL DIVISOR
0007 B9 00230 CP C ; ES SUPERIOR A AL DIVISOR?
0008 3802 00240 JR C,H3 ; NO, PASAR AL BIT SIGUIENTE
000A 91 00250 H4 SUB C ; SUSTRAE EL DIVISOR
000B 2C 00260 INC L ; 1 EN EL COCIENTE
000C 10FB 00270 H3 DJNZ H2 ; BUCLE PARA TRATAR TODOS LOS BITS
000E C9 00280 RET ; FIN DEL SUBPROGRAMA
0000 00290 END
0000 TOTAL ERRORES

```

El programa efectúa una división de un número de 16 bits por un número de 8 bits, coloca el cociente en un número de 16 bits y el resto en un número de 8 bits. El único caso en el que puede producirse un desbordamiento es cuando el divisor (C) es nulo.

### 3.3. Generador de números aleatorios: RND

Este subprograma calcula un número aleatorio de 8 bits cuyo valor está comprendido entre 1 y un número máximo suministrado cuando se realiza la llamada al subprograma. Este empieza por determinar un número aleatorio comprendido entre 0 y 255, utilizando el registro REFRESH del Z 80 y la variable SEED modificada a cada

llamada. El registro R contiene un número pseudoaleatorio comprendido entre 0 y 127; de este modo hay que efectuar un pequeño tratamiento para obtener un número que varíe entre 0 y 255. Conocido este número se le multiplica por el número máximo dado al entrar. Guardando solamente el octeto de más peso del resultado, al que se le añade 1, obtenemos el número aleatorio deseado. La función RANDOM debe ser llamada al principio del programa para inicializar la variable SEED.

```

00010 .....
00020 .....
00030 : INICIALIZACION DEL GENERADOR ALEATORIO
00040 : FUNCION RANDOM
00050 .....
00060 .....
00070 .....
00080 .....
00090 .....
000A ED5F 00080 RANDOM LD A,R ; NUMERO ALEATORIO
000B C9 00090 LD (SEED),A ; INICIALIZACION DEL SEED
00100 RET
00110 .....
00120 .....
00130 .....
00140 .....
00150 .....
00160 .....
00170 .....
00180 .....
00190 .....
00200 .....
00210 .....
00220 .....
00230 .....
00240 .....
0006 D6 00250 RND PUSH DE ; RESGUARDA LOS REGISTROS DE
0007 E5 00260 PUSH HL ; Y HL
0008 1600 00270 LD D,0 ; EL NUMERO MAXIMO ES COLOCADO
000A 8F 00280 LD E,A ; EN DE
000B ED5F 00290 LD A,R ; LECTURA DE UN NUMERO ALEATORIO
00300 .....
000D 6F 00310 LD LA ; ESTE NUMERO ES RESGUARDADO EN L
000E 3A2E00 00320 LD A,(SEED); LECTURA DEL SEED PRECEDENTE
0011 AD 00330 XOR L ; CALCULO DEL SEED SIGUIENTE
0012 17 00340 RLA ; A VARIA ENTRE 0 Y 254
0013 6F 00350 LD LA ; RESGUARDA EN L
0014 ED5F 00360 LD A,R ; NUMERO ALEATORIO
0016 AD 00370 XOR L ; CALCULO DEL SEED SIGUIENTE
0017 322E00 00380 LD (SEED),A; NUEVO SEED
001A CD2200 00390 CALL MUL ; MULTIPLICAR EL NUMERO ALEATORIO
00400 : POR EL NUMERO MAXIMO DADO
001D 7C 00410 LD A,H ; NUMERO ALEATORIO ENTRE 0 Y
00420 : EL NUMERO MAXIMO MENOS 1
001E 3C 00430 INC A ; NUMERO ALEATORIO ENTRE 1 Y
00440 : EL NUMERO MAXIMO
001F E1 00450 POP HL ; RESTAURACION DE LOS REGISTROS
0020 D1 00460 POP DE ; DE Y HL
0021 C9 00470 RET
00480 .....
00490 .....
00500 .....
00510 .....
0022 210000 00520 MUL LD HL,0
0025 29 00530 H0 ADD HL,HL
0026 CB27 00540 SLA A
0028 3001 00550 JR NCH1
002A 19 00560 ADD HL,DE
002B 20FB 00570 H1 JR NZ,H0
002D C9 00580 RET
0001 00590 SEED DEFB 1
0000 00600 END
0000 TOTAL ERRORES

```

### 3.4. Conversión de un número binario entero en una serie de caracteres ASCII: TRF

Este subprograma será muy útil para visualizar las puntuaciones en la pantalla de un programa de juego. Cada vez que la puntuación sea modificada se le llamará para reescribir la nueva puntuación, esta rutina emplea un procedimiento simple, pero que tiene la ventaja de ser muy rápido. Determina sucesivamente todas las cifras del número a traducir, comenzando por la cifra más significativa. Esto se obtiene dividiendo sucesivamente el número por todas las potencias de 10, empezando por 10000, llegando hasta 10. El cociente transformado en ASCII da una de las cifras deseadas, el resto se utiliza como nuevo argumento para efectuar las otras divisiones.

Número = cifra 1 \* 10000 + resto 1  
 resto 1 = cifra 2 \* 1000 + resto 2  
 resto 2 = cifra 3 \* 100 + resto 3  
 resto 3 = cifra 4 \* 10 + resto 4  
 resto 4 = cifra 5

Las cifras así calculadas se colocan las unas detrás de las otras en una zona reservada llamada BUF. Después de la última división se ordena el número de unidades, se coloca un cero al final de la cadena y se posiciona HL al principio. Entonces se puede utilizar el subprograma MES para la visualización en la pantalla (véase cap. 4).

```

00010 : .....
00020 :
00030 : TRANSFORMACION DE UN NUMERO ENTERO CON SIGNO SOBRE 16
00040 : BITS. EN UNA SERIE DE CARACTERES ASCII QUE LO REPRESENTAN.
00050 : EL NUMERO PUEDE VARIAR ENTRE -32768 y 32767
00060 :
00070 : PARA OBTENER UNA CONVERSION SOBRE UN ENTERO SIN SIGNO
00080 : CUYO VALOR ESTA COMPRENDIDO ENTRE 0 y 65535 ES
00090 : SUFICIENTE DESTRUIR LAS LINEAS 230 A 300
00100 :
00110 : ENTRADA: HL CONTIENE EL NUMERO A CONVERTIR
00120 :
00130 : SALIDA: HL CONTIENE LA DIRECCION DEL PRIMER
00140 : ELEMENTO DE LA CADENA QUE REPRESENTA
00150 : AL NUMERO. ASI ES POSIBLE UTILIZAR
00160 : EL SUBPROGRAMA MES PARA LA VISUALIZACION
00170 : .....
00180 :
    
```

```

0000 DD218000 00180 ;
0000 DD218000 00200 TRF LD IX,BUF+1 ; IX SIRVE PARA LLENAR EL BUFFER
0004 DD36FF20 00210 ; QUE CONTENDRA LA CADENA
0008 CB7C 00220 LD (IX-1),20H ; NO HAY SIGNO POR DEFECTO
000A 280B 00230 BIT 7,H ; EL NUMERO ES NEGATIVO?
000C 110000 00240 JR Z,TRD ; SI NO
000F EB 00250 LD DE,0 ; PARA SUSTRAEER EL HL DE 0
0010 B7 00260 EX DE,HL ;
0011 ED62 00270 OR A ; PUESTA A 0 DEL CARRY
0013 DD36FF2D 00280 SBC HL,DE ; HACER 0-HL PARA OBTENER EL
0017 111027 00290 ; VALOR ABSOLUTO DE HL
001A CD3D00 00300 LD (IX-1),'-' ; PONER UN SIGNO MENOS EN CABEZA
001D 11E803 00310 TRD LD DE,10000 ;
0020 CD3D00 00320 CALL CAR ; GUARDAR EL NUMERO DE DECENAS
0023 116400 00330 ; DE MIL
0028 CD3D00 00340 LD DE,1000 ; GUARDAR EL NUMERO DE LOS MILLARES
002B 116400 00350 CALL CAR ;
002E 110A00 00360 LD DE,100 ; GUARDAR EL NUMERO DE LAS CENTENAS
0032 CD3D00 00370 CALL CAR ;
0035 00380 LD DE,10 ; GUARDAR EL NUMERO DE LAS UNIDADES
0039 214F00 00390 CALL CAR ; L CONTIENE EL NUMERO DE LAS UNIDADES
003C C9 00400 LD A,L ; TRANSFORMAR EN ASCII
003D B7 00410 ADD A,30H ; PUESTA EN EL BUFFER
003E 00FF 00420 LD (IX+0),A ; PONER UN 0 AL FINAL
0040 ED62 00430 LD (IX+1),0 ; HL APUNTA HACIA EL PRIMERO
0043 00440 LD HL,BUF ; ELEMENTO
0046 00460 RET ;
00470 : .....
00480 :
00490 : SUBPROGRAMA QUE DIVIDE HL POR DE COLOCANDO
00500 : EL COCIENTE +30H EN EL BUFFER
00510 : .....
00520 :
00530 :
00540 CAR OR A ; PUESTA A 0 DEL CARRY
00550 LD B,255 ; B=-1
00560 G9 SBC HL,DE ; SE SUSTRAE (DE) DE (HL) HASTA QUE
00570 ; HAYA DESBORDAMIENTO
00580 INC B ; INCREMENTAR EL COCIENTE
00590 JR NC,G9 ; SI NO HAY DESBORDAMIENTO
005A 19 00600 ADD HL,DE ; HL CONTIENE EL RESTO
005B 78 00610 LD A,B ; A CONTIENE EL COCIENTE
005C C830 00620 ADD A,30H ; TRANSFORMACION EN ASCII
005D DD7700 00630 LD (IX+0),A ; PUESTA EN EL BUFFER
005E DD23 00640 INC IX ; INCREMENTAR EL PUNTERO DEL BUFFER
005F C9 00650 RET ;
0060 00660 BUF DEFS 7 ; BUFFER CONTENIENDO LA CADENA
0061 00670 END ;
0062 00680
0063 00690
0064 00700
0065 00710
0066 00720
0067 00730
0068 00740
0069 00750
0070 00760
0071 00770
0072 00780
0073 00790
0074 00800
0075 00810
0076 00820
0077 00830
0078 00840
0079 00850
0080 00860
0081 00870
0082 00880
0083 00890
0084 00900
0085 00910
0086 00920
0087 00930
0088 00940
0089 00950
0090 00960
0091 00970
0092 00980
0093 00990
0094 01000
0095 01010
0096 01020
0097 01030
0098 01040
0099 01050
0100 01060
0101 01070
0102 01080
0103 01090
0104 01100
0105 01110
0106 01120
0107 01130
0108 01140
0109 01150
0110 01160
0111 01170
0112 01180
0113 01190
0114 01200
0115 01210
0116 01220
0117 01230
0118 01240
0119 01250
0120 01260
0121 01270
0122 01280
0123 01290
0124 01300
0125 01310
0126 01320
0127 01330
0128 01340
0129 01350
0130 01360
0131 01370
0132 01380
0133 01390
0134 01400
0135 01410
0136 01420
0137 01430
0138 01440
0139 01450
0140 01460
0141 01470
0142 01480
0143 01490
0144 01500
0145 01510
0146 01520
0147 01530
0148 01540
0149 01550
0150 01560
0151 01570
0152 01580
0153 01590
0154 01600
0155 01610
0156 01620
0157 01630
0158 01640
0159 01650
0160 01660
0161 01670
0162 01680
0163 01690
0164 01700
0165 01710
0166 01720
0167 01730
0168 01740
0169 01750
0170 01760
0171 01770
0172 01780
0173 01790
0174 01800
0175 01810
0176 01820
0177 01830
0178 01840
0179 01850
0180 01860
0181 01870
0182 01880
0183 01890
0184 01900
0185 01910
0186 01920
0187 01930
0188 01940
0189 01950
0190 01960
0191 01970
0192 01980
0193 01990
0194 02000
    
```

### 3.5. Conversión de un número dado bajo la forma de una serie de caracteres ASCII en un número binario: ASCBIN

Este programa efectúa la transformación inversa de la precedente. Convierte un número dado bajo forma de una serie de caracteres ASCII en su representación binaria de 16 bits.

Ejemplo:

Serie ASCII «4 1 8» valor binario: 0000000110100010  
 códigos correspondientes «34 31 38» valor hexadecimal: 0 1 A 2  
 en hexadecimal

El programa siguiente utiliza el hecho de que un número decimal que se escribe *xyzt* en ASCII vale:  $((x \cdot 10) + y) \cdot 10 + z) \cdot 10 + t$ . El algoritmo informático utilizado sale directamente de esta fórmula.

Hay que hacer notar que la multiplicación por 10 se realiza mediante cuatro sumas.

```

00010 : .....
00020 :
00030 : Transformación de un número positivo entero
00040 : dado bajo forma de una serie de caracteres
00060 : ASCII en su valor binario sobre 16 bits
00080 :
00070 : Entrada: DE contiene la dirección de la serie
00080 : de caracteres ASCII constituida por
00090 : cifras y terminada por un símbolo
00100 : ASCII distinto de un número
00110 :
00120 : Salida: HL contiene el número de 16 bits resultante
00130 :
00140 : AF, BC, DE, HL son modificados
00160 : .....
00170 :
0000 210000 00180 ASCBIN LD HL,0 ; Resultado nulo a priori
0003 1A 00190 HB LD A,(DE) ; Lectura carácter ASCII
0004 D630 00200 SUB '0' ; Conversión para obtener su valor
0006 D8 00210 RET C ; Retorno si no es una cifra
0007 FE0A 00220 CP 10
0009 D0 00230 RET NC ; Retorno si no es una cifra
000A 13 00240 INC DE ; Incrementar puntero
000B 44 00250 LD B,H ; Copiar HL en BC
000C 4D 00260 LD C,L
000D 29 00270 ADD HL,HL ; Resultado = resultado . 2
000E 29 00280 ADD HL,HL ; Resultado = resultado . 4
000F 09 00290 ADD HL,BC ; Resultado = resultado . 8
0010 29 00300 ADD HL,HL ; Resultado = resultado . 10
0011 0600 00310 LD B,0
0013 4F 00320 LD C,A ; Cifra en BC
0014 09 00330 ADD HL,BC ; Añadir valor de la cifra
0015 18EC 00340 JR HB ; Pasar a la cifra siguiente
0000 00360 END
  
```

## 4. Las entradas/salidas

En este capítulo estudiaremos la programación de las entradas/salidas, es decir, la gestión software de los diversos periféricos de que dispone el ZX SPECTRUM. Este es un punto fundamental de la programación en ensamblador ya que es el medio utilizado para visualizar los resultados y conseguir informaciones.

Examinaremos sucesivamente la interfase entre los periféricos más corrientes y la unidad central; también daremos ejemplos de los subprogramas que permiten controlar a estos periféricos. Al igual que en el capítulo anterior, estos subprogramas se situarán en la dirección cero.

### 4.1. La pantalla de visualización

#### 4.1.1. Distribución de la pantalla

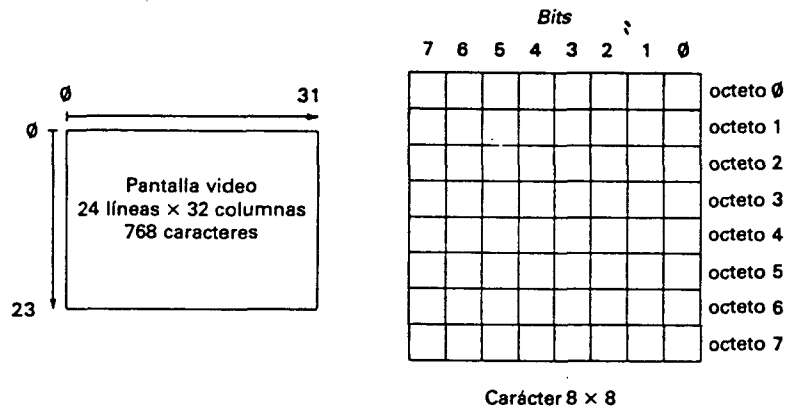
Para que puedan realizarse las operaciones de visualización, el ZX SPECTRUM posee una memoria viva de 6,75 K que sirve de interfase entre la pantalla y la unidad central. Esta memoria es leída de forma cíclica por un dispositivo hardware que genera las señales eléctricas necesarias para la visualización en pantalla de las informaciones contenidas en la memoria. Además, la escritura en esta memoria produce como consecuencia inmediata la visualización de puntos sobre la pantalla o la selección de un color.

Esta memoria, a la que llamaremos *memoria video*, está constituida por dos partes:

La *primera parte* contiene la lista de los puntos de la pantalla. Ocupa 6 K y se sitúa entre las direcciones 4000H y 57FFH, ambas inclusive. A cada punto de la pantalla, o *pixel*, se le asocia un bit de esta zona de memoria. Si el bit vale 1, el punto se visualiza con el color de la tinta (orden INK del Basic). Si el bit vale 0, el punto se visualiza con el color del fondo (orden PAPER del Basic). Supongamos que el color del fondo es blanco y el color de la tinta negro. El pixel aparece

en negro sobre fondo blanco. Así, colocando algunos bits de esta zona de memoria al valor 1, provocamos la visualización de pequeños puntos negros en la pantalla.

La pantalla se divide en 24 líneas de 32 caracteres. Convencionalmente las líneas son numeradas de 0 a 23. La línea 0 está arriba de la pantalla y la línea 23 abajo. Las columnas se numeran de izquierda a derecha con los números enteros comprendidos entre 0 y 31. Cada carácter está constituido por una matriz de 8 posiciones por 8, o sea 64 posiciones. Esta matriz se almacena en memoria mediante 8 octetos. Cada octeto contiene una línea de esta matriz. Cada uno de los bits de este octeto representa una posición de la matriz. El bit más elevado contiene el punto que está más a la izquierda de la línea, el bit más bajo contiene el punto más a la derecha.

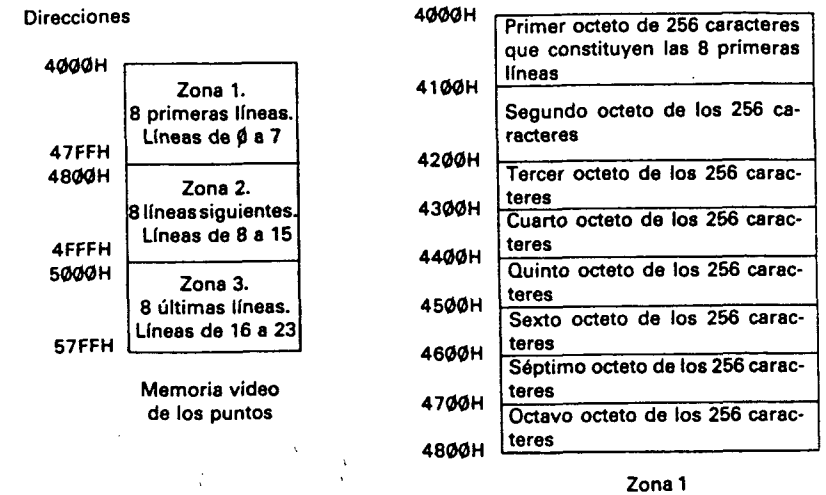


La resolución del SPECTRUM es por lo tanto de  $24 \cdot 8 = 192$  de  $32 \times 8 = 256$  puntos, lo que es muy aceptable para un microordenador de esta talla.

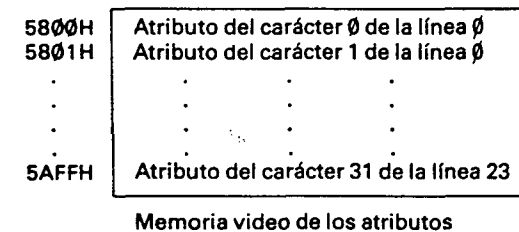
La posición en la memoria video de los 8 octetos que contienen el carácter es menos fácil de comprender.

La memoria video que contiene los puntos se divide en tres partes de 2 K octetos, situados respectivamente en las direcciones 4000H, 4800H y 5000H. La primera parte contiene las ocho primeras líneas de la pantalla, la segunda parte las ocho siguientes y la tercera parte las ocho últimas líneas de la pantalla. En cada parte, los 256 primeros octetos contienen los primeros octetos de cada uno de los 256 caracteres ( $8 \times 32$ ), constituyendo las ocho líneas en el orden en el que estos caracteres son visualizados en la pantalla (los 32 primeros octetos contienen los primeros octetos de la primera línea, el primer octeto

corresponde al carácter de la izquierda de la línea). Los 256 octetos siguientes contienen los segundos octetos de los 256 caracteres que constituyen las ocho líneas y así sucesivamente hasta los 256 últimos octetos que contienen los octavos octetos de los 256 caracteres.



La segunda parte de la memoria video contiene los atributos gráficos asociados a cada carácter. Esto ocupa 768 octetos y se sitúa en las direcciones 5800H y 5AFFH. A cada carácter le corresponde un octeto de atributos, colocado en esta memoria en el orden en el que estos caracteres aparecen en la pantalla (los 32 primeros octetos contienen la primera línea, el primer octeto corresponde al carácter de la izquierda de la línea).

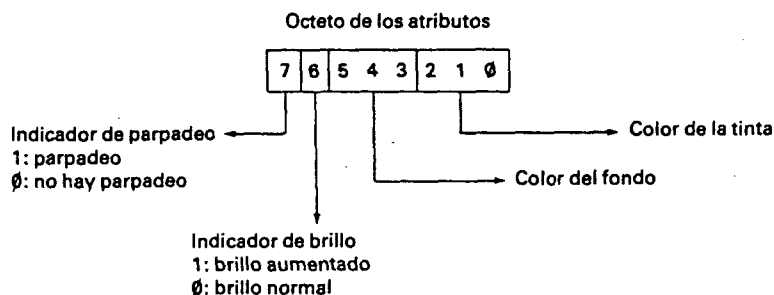




El ZX SPECTRUM puede visualizar ocho colores indicados con los números de 0 a 7. En consecuencia, un color puede ser memorizado en tres bits según la tabla siguiente:

Código del color	Colores
0 0 0	Negro
0 0 1	Azul
0 1 0	Rojo
0 1 1	Magenta
1 0 0	Verde
1 0 1	Cyan
1 1 0	Amarillo
1 1 1	Bianco

El octeto de atributos asociado a un carácter memoriza el color del fondo (color de los puntos, donde el bit asociado en memoria video es nulo) en tres bits, el color de la tinta (color de los puntos cuyo bit asociado vale 1) en tres bits, un indicador de brillo mediante 1 bit (vale 1 si el carácter tiene más brillo) y un indicador de parpadeo en 1 bit (vale 1 si el carácter parpadea).



Este procedimiento de codificación de colores relativos a un carácter limita las posibilidades gráficas del ZX SPECTRUM, puesto que no se puede fijar individualmente el color de cada uno de los puntos. No obstante, un carácter podrá colorearse rápidamente poniendo el color de fondo y el color de la tinta del color deseado, y esto sin tocar los puntos contenidos en el carácter. Este método de coloreado se utilizará en los juegos de acción rápida para el dibujo del decorado.

El último elemento programable en pantalla es el color del borde de la misma (orden BORDER del Basic). Este color, que puede ser escogido entre los ocho colores precedentes, se fija enviando el código

del color al port 254. Para ello podrá utilizarse la línea de instrucciones siguientes:

```
LD  A, COLOR      ; 0 ≤ COLOR ≤ 7
OUT (254), A      ; fija el color del borde
```

El port 254 es un port cuyo número está sobre 8 bits. No hay pues que colocar el octeto de más peso de este número en el registro A.

La memoria video, que ocupa 6,75 K en memoria viva, limita la memoria viva utilizable por los programas Basic y también los programas en lenguaje máquina. Este último contiene un poco más de 9 K octetos en un SPECTRUM 16 K, lo cual no permite el funcionamiento de programas grandes. En ensamblador habrá que vigilar esta memoria video situando los programas más allá de la dirección 5B00H. Si desea utilizar los subprogramas del interpretador Basic sitúe su programa más allá de la dirección 5CB6H.

Examinemos ahora diferentes subprogramas de visualización utilizando la organización de la pantalla tal como acabamos de describir.

#### 4.1.2. Borrado de la pantalla: CLS

Este subprograma borra la pantalla colocando a cero todos los puntos de la memoria video y colocando a 38H (tinta negra sobre fondo blanco en brillo normal y sin destello) todos los atributos de los caracteres de la pantalla.

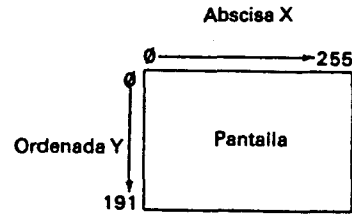
```
00010 .....
00020 .....
00030 ..... BORRADO DE LA PANTALLA
00040 .....
00080 .....
00080 .....
0000 210040 00070 CLS LD HL,4000H ; INICIO DE LA MEMORIA DE LA PANTALLA
0003 110140 00080 LD DE,4001H ; INICIO DE LA MEMORIA DE LA PANTALLA +1
0008 01FF17 00090 LD BC,17FFH ; TAMAÑO DE LA MEMORIA DE PANTALLA - 1
0009 3800 00100 LD (HL),0 ; BORRADO DEL PRIMER CARACTER
000B ED80 00110 LDIR ; BORRADO DE TODA LA PANTALLA
000D 23 00120 INC HL ; INICIO ZONA DE ATRIBUTOS
000E 13 00130 INC DE ; INICIO ZONA DE ATRIBUTOS + 1
000F 01FF02 00140 LD BC,2FFH ; LONGITUD ZONA DE ATRIBUTOS - 1
0012 3838 00150 LD (HL),38H ; NEGRO SOBRE FONDO BLANCO
0014 ED80 00160 LDIR ; ESCRITURA DE TODOS LOS ATRIBUTOS
0016 C9 00170 RET
0000 00180 END
```

#### 4.1.3. Subprogramas gráficos: SET, RESET y POINT

El subprograma SET visualiza un punto en la pantalla, RESET apaga un punto en la pantalla y POINT comprueba si un punto está presente en la misma. El punto es descrito por su abscisa, comprendi-

da entre 0 y 255 y su ordenada, comprendida entre 0 y 191. Estos tres subprogramas llaman a un mismo subprograma llamado CONV que calcula la dirección de la memoria video a la que debe accederse para realizar la función deseada. También suministra un octeto cuyo único bit a 1 corresponde al bit al que hay que acceder en la posición memoria situada en la dirección calculada.

Estos subprogramas no actúan sobre los atributos que deberán ser convenientemente posicionados para que la visualización corresponda a lo que se desea.



```

00010 :-----
00020 :
00030 : FUNCIONES GRAFICAS: SET, RESET Y PUNTO
00040 :
00050 : ENTRADA: H CONTIENE LA ORDENADA DEL PUNTO
00060 :          GRAFICO 0 <= H <= 191
00070 :          L CONTIENE LA ABCISIA DEL PUNTO GRAFICO
00080 :          0 <= L <= 255
00090 :
00100 : TODOS LOS REGISTROS SON RESGUARDADOS SALVO AF
00110 :
00120 :-----
00130 :
00140 :
00150 :-----
00160 :
00170 : ENCENDIDO DE UN PUNTO
00180 :-----
00190 :
0000 E5 00200 SET  PUSH  HL      ; GUARDAR POSICION
0001 CD1800 00210 CALL  CONV  ; BUSQUEDA DE LA DIRECCION DE PANTALLA
0004 B8 00220 OR    (HL)  ; PUESTA A 1 DE UN BIT
0005 77 00230 LD    (HL),A ; VISUALIZACION
0006 E1 00240 POP  HL      ; RESTAURAR POSICION
0007 C9 00250 RET

00260 :-----
00270 :
00280 : DESAPARICION DE UN PUNTO
00290 :-----
00300 :
00310 :
0008 E5 00320 RESET PUSH  HL      ; GUARDAR POSICION
0009 CD1800 00330 CALL  CONV  ; BUSQUEDA DE LA DIRECCION DE PANTALLA
000C 2F 00340 CPL
000D A8 00350 AND  (HL)  ; PUESTA A CERO DE UN BIT
000E 77 00360 LD    (HL),A ; VISUALIZACION
000F E1 00370 POP  HL      ; RESTAURAR POSICION
0010 C9 00380 RET

```

```

00390 :-----
00400 :
00410 : COMPROBACION DE UN PUNTO
00420 :-----
00430 : EL INDICADOR Z ESTA A 1 SI EL PUNTO ESTA APAGADO
00440 : EL INDICADOR Z ESTA A 0 SI EL PUNTO ESTA ENCENDIDO
00450 :-----
00460 :
00470 :
0011 E5 00480 PUNTO PUSH  HL      ; GUARDAR POSICION
0012 CD1800 00490 CALL  CONV  ; BUSQUEDA DIRECCION DE PANTALLA
0015 A8 00500 AND  (HL)  ; COMPROBAR SI EL BIT ESTA A UNO
0016 E1 00510 POP  HL      ; RESTAURAR LA POSICION
0017 C9 00520 RET

00530 :-----
00540 :
00550 : SUBPROGRAMA DE CONVERSION
00560 :-----
00570 : ENTRADA: HL = ORDENADA DEL PUNTO
00580 :          L = ABCISIA DEL PUNTO
00590 :
00600 : SALIDA: HL = DIRECCION DEL OCTETO CORRESPONDIENTE AL PUNTO
00610 :          A = VALOR DEL OCTETO
00620 :-----
00630 :
00640 :
0018 C5 00650 CONV  PUSH  BC      ; RESGUARDAR BC
0019 7C 00660 LD    AH      ; ORDENADA DEL PUNTO
001A 2640 00670 LD    H,40H  ; INICIO PRIMERA ZONA DE LA PANTALLA
001C D640 00680 SUB  64      ; QUITAR LONGITUD DE UNA ZONA
001E 380A 00690 JR    C,ZA   ; ¿PRIMER ZONA?
0020 2648 00700 LD    H,48H  ; PRINCIPIO SEGUNDA ZONA DE LA PANTALLA
0022 D640 00710 SUB  64      ; QUITAR LONGITUD DE UNA ZONA
0024 3804 00720 JR    C,ZA   ; ¿SEGUNDA ZONA?
0026 2660 00730 LD    H,60H  ; INICIO ULTIMA ZONA DE LA PANTALLA
0028 D640 00740 SUB  64      ; QUITAR LONGITUD DE UNA ZONA
002A C640 00750 ZA  ADD  A,64   ; POSICION EN LA ZONA
002C 4F 00760 LD    CA      ; GUARDAR ESTE VALOR EN C
002D E607 00770 AND  7      ; DECALADO VERTICAL EN EL CARACTER
002F 84 00780 ADD  AH      ; OCTETO DE MAS PESO DE
0030 67 00790 LD    HA      ; LA DIRECCION VIDEO
0031 79 00800 LD    AC      ; RECUPERAR POSICION EN LA ZONA
0032 E638 00810 AND  38H    ; NUMERO DE LINEA . 8
0034 07 00820 RLCA   ; NUMERO DE LINEA . 16
0035 07 00830 RLCA   ; NUMERO DE LINEA . 32
0036 4F 00840 LD    CA      ; GUARDAR EN C
0037 7D 00850 LD    AL      ; ABCISIA DEL PUNTO
0038 0F 00860 RRCA   ; DIVISION POR TRES PARA OBTENER
0039 0F 00870 RRCA   ; EL NUMERO DEL CARACTER EN
003A 0F 00880 RRCA   ; LA LINEA
003B E61F 00890 AND  1FH    ; POSICION DEL CARACTER EN LA LINEA
003D 81 00900 ADD  AC      ; AÑADIR LAS LINEAS PRECEDENTES
003E 4D 00910 LD    CL      ; GUARDAR ABCISIA EN C
003F 6F 00920 LD    LA      ; OCTETO DE MENOS PESO DE LA DIRECCION
0040 79 00930 LD    AC      ; RESTURAR ABCISIA
0041 E607 00940 AND  7      ; DECALADO HORIZONTAL EN EL CARACTER
0043 3C 00950 INC  A      ; MAS UNO
0044 4F 00960 LD    CA      ; CONTADOR DE BUCLE
0045 3E01 00970 LD    A,I    ;
0047 0F 00980 ZB  RRCA   ; DECALAR A HASTA OBTENER EL OCTETO
0048 0D 00990 DEC  C      ; QUE POSEE EL BIT A 1
0049 20FC 01000 JR    NZ,ZB  ; LA POSICION DESEADA
004B C1 01010 POP  BC      ; RESTAURAR BC
004C C9 01020 RET
0000 01030 END

```

#### 4.1.4. Trazado de una recta: RECTA

Este programa permite que, sobre la pantalla, pueda trazarse cualquier recta definida por sus coordenadas iniciales ( $X_1$  e  $Y_1$ ) y sus coordenadas finales ( $X_2$  e  $Y_2$ ). Para explicar el algoritmo empleado supondremos que la variación de  $Y$  es inferior a la variación de  $X$ , o sea:

$$|Y_2 - Y_1| < |X_2 - X_1| \text{ y que } X_1 < X_2; Y_1 < Y_2$$

Se utiliza un bucle para poder dibujar sucesivamente todos los puntos de la recta. La abscisa del punto actual se incrementa a cada paso. Contrariamente, la ordenada se incrementa cada vez que es necesario para conservar la dirección de la recta. Todo el problema consiste en saber cuándo hay que incrementarla. Para ello examinaremos la ecuación de la recta:

$$Y = Y_1 + (X - X_1) \frac{Y_2 - Y_1}{X_2 - X_1}$$

Sea  $X_i$  la abscisa del punto actual de la recta y  $X_{i+1}$  la abscisa siguiente:

$$\text{Tenemos: } X_{i+1} = X_i + 1$$

$$\text{y } Y_{i+1} = Y_1 + (X_i + 1 - X_1) \frac{Y_2 - Y_1}{X_2 - X_1}$$

$$Y_{i+1} = Y_i + \frac{Y_2 - Y_1}{X_2 - X_1}$$

Entre dos puntos la ordenada es aumentada en  $\frac{Y_2 - Y_1}{X_2 - X_1}$  que es inferior a uno, según nuestra hipótesis. Si despreciamos este término no aumentando  $Y$ , cometemos un error de  $\frac{Y_2 - Y_1}{X_2 - X_1}$ . Así, en el siguiente punto tendremos:

$$Y_i + 2 = Y_i - 2 \cdot \frac{Y_2 - Y_1}{X_2 - X_1}$$

Si no aumentamos  $Y$  durante  $n$  puntos, el error acumulado será igual a  $n \cdot \frac{Y_2 - Y_1}{X_2 - X_1}$ . Cuando este valor sea superior a  $0,5$ , será necesario incrementar la ordenada. El procedimiento utilizado en el programa se deriva de lo que acabamos de decir. Se controla una

variable llamada ERR que no contiene el error propiamente dicho, que es un número fraccionario, sino un número entero igual a:

$n \cdot (Y_2 - Y_1) + \frac{X_2 - X_1}{2}$ . El error será superior a  $0,5$  si este número sobrepasa  $X_2 - X_1$ .

$$n \cdot (Y_2 - Y_1) + \frac{X_2 - X_1}{2} > X_2 - X_1 \Rightarrow n \cdot \frac{(Y_2 - Y_1)}{X_2 - X_1} > 0,5$$

Entonces habrá que incrementar la ordenada y sustraer el valor  $X_2 - X_1$  a la variable ERR, que viene a ser lo mismo que sustraer 1 del error. De esta forma se obtiene el trazado completo de la recta haciendo variar  $X_i$  desde  $X_1$  a  $X_2$ .

El programa general tiene una fase de inicialización destinada a calcular  $|X_2 - X_1|$  e  $|Y_2 - Y_1|$  y a determinar el incremento que hay que añadir a  $X$  y a  $Y$  ( $-1$  o  $+1$ ) según el signo de  $X_2 - X_1$  y de  $Y_2 - Y_1$ . Encontramos a continuación dos bucles para el trazado de la recta correspondiente a uno de los casos:  $|X_2 - X_1| > |Y_2 - Y_1|$  y  $|X_2 - X_1| < |Y_2 - Y_1|$ .

Entonces es posible trazar cualquier recta mientras sus coordenadas iniciales y finales estén comprendidas entre los límites admitidos y que sean diferentes. Al igual que para las funciones gráficas, antes de trazar la recta se procederá a posicionar los atributos gráficos.

```

00010 .....
00020 .....
00030 : TRAZADO DE UNA RECTA EN LA PANTALLA
00040 .....
00050 : ENTRADA: D CONTIENE LA ORDENADA INICIAL Y1
00060 : E CONTIENE LA ABCISCA INICIAL X1
00070 : H CONTIENE LA ORDENADA FINAL Y2
00080 : L CONTIENE LA ABCISCA FINAL X2
00090 .....
00100 .....
00110 .....
0000 0601 00120 RECTA LD B,1 ; EL INCREMENTO EN Y VALE 1 POR DEFECTO
0002 48 00130 LD C,B ; EL INCREMENTO EN X VALE 1 POR DEFECTO
0003 7A 00140 LD A,D ; A = Y2
0004 94 00150 SUB H ; A = Y2 - 1
0005 3004 00160 JR NC,GO ; SI Y2 - Y1 >= 0 NO HACER NADA
0007 06FF 00170 LD B,OFFH ; SI NO TOMAR -1 COMO INCREMENTO
0009 7C 00180 LD A,H ;
.....
000A 92 00190 SUB D ; A = -(Y2 - Y1)
000B 57 00200 GO LD D,A ; D = ABS(Y2 - Y1)
000C 7B 00210 LD A,E ; A = X2
000D 98 00220 SUB L ; A = X2 - X1
000E 3004 00230 JR NC,G1 ; SI X2 - X1 >= 0 NO HACER NADA
0010 0EFF 00240 LD C,OFFH ; SI NO TOMAR -1 COMO INCREMENTO
0012 7D 00250 LD A,L ;
0013 83 00260 SUB E ; A = -(X2 - X1)
0014 5F 00270 G1 LD E,A ; E = ABS(X2 - X1)
0015 ED436100 00280 LD (G3),BC ; RESGUARDAR LOS INCREMENTOS EN G3
0019 7A 00290 LD A,D ; A = ABS(Y2 - Y1)

```

```

001A BB      00300      CP      E      ; ¿ES ABS (Y2 - Y1) >= ABS (X2 - X1)?
001B 3021    00310      JR      NC,G2 ; SI ES SI PASAR AL SEGUNDO BUCLE
001D 43      00320      LD      B,E   ; B = ABS (X2 - X1): CONTADOR DE BUCLE
001E 4B      00330      LD      C,E   ; C = ERR
001F CB39    00340      SRL     C     ; ERR = ABS (X2 - X1)/2
0021 CD6300  00350 G8      CALL    SET   ; VISUALIZACION DEL SEGMENTO H.L
0024 3A6100  00360      LD      A,(G3); INCREMENTO EN X
0027 86      00370      ADD     A,L   ; AÑADIR EL INCREMENTO A XI
0028 6F      00380      LD      LA   ; NUEVO XI
0029 79      00390      LD      AC   ; A = ERR
002A 82      00400      ADD     A,D   ; A = ERR + ABS (Y2 - Y1)
002B 4F      00410      LD      CA   ; ERR = ERR + ABS (Y2 - Y1)
002C 3806    00420      JR      C,ZC ; ERR >= 256
002E 93      00430      SUB     E     ; SUSTRAR ABS (X2 - X1)
002F 3809    00440      JR      C,G4 ; SI ERR < ABS (X2 - X1) PASAR AL PUNTO
0031 1801    00450      JR      ZD   ; SIGUIENTE
0033 93      00460 ZC      SUB     E     ; SUSTRAR ABS (X2 - X1)
0034 4F      00470 ZD      LD      CA   ; ERR = ERR - ABS (X2 - X1)
0036 3A6200  00480      LD      A,(GE + 1); INCREMENTO EN Y
0038 84      00490      ADD     A,H   ; INCREMENTAR Y
0039 67      00500      LD      HA   ; NUEVA Y
003A 10E6    00510 G4      DJNZ   G6   ; BUCLE PARA VISUALIZAR TODOS LOS
003C 181F    00520      JR      G6   ; PUNTOS
003E 42      00530      JR      G6   ; VISUALIZAR EL PUNTO FINAL
003F 4A      00540 G2      LD      B,D   ; B = ABS (Y2 - Y1): CONTADOR DE BUCLE
0040 CB39    00550      LD      C,D   ; ERR = C
0042 CD6300  00560      SRL     C     ; ERR = ABS (Y2 - Y1)/2
0045 3A6200  00570 G7      CALL    SET   ; VISUALIZACION DE UN SEGMENTO
0048 84      00580      LD      A,(GE + 1); INCREMENTO EN Y
0049 67      00590      ADD     A,H   ; INCREMENTAR Y
004A 79      00600      LD      HA   ; NUEVA Y
004B 83      00610      LD      AC   ; A = ERR
004C 4F      00620      ADD     A,E   ; A = ERR + ABS (X2 - X1)
004D 3806    00630      LD      CA   ; ERR = A
004F 92      00640      JR      C,ZE ; ERR >= 256
0050 3909    00650      SUB     D     ; SUSTRAR ABS (Y2 - Y1)
0052 1801    00660      JR      C,G8 ; SI ERR < ABS (Y2 - Y1) PASAR AL PUNTO
0054 92      00670      JR      ZH   ; SIGUIENTE
0055 4F      00680 ZE      SUB     D     ; SUSTRAR ABS (Y2 - Y1)
0056 4F      00690 ZH      LD      CA   ; ERR = ERR - ABS (Y2 - Y1)
0058 85      00700      LD      A,(G3); INCREMENTO EN X
005A 6F      00710      ADD     A,L   ; INCREMENTAR X
005B 10E6    00720      LD      LA   ; NUEVA X
005D CD6300  00730 G8      DJNZ   G7   ; BUCLE PARA VISUALIZAR TODOS
0060 C9      00740      LD      G7   ; LOS PUNTOS
0062          00750      CALL    SET ; VISUALIZACION DEL ULTIMO PUNTO
0063          00760      RET     ;
0065          00770 G3      DEFB   2     ;
0066          00780 SET   EQU    $     ; EL SUBPROGRAMA SET DEBERA
0067          00790      EQU    $     ; IMPLANTARSE AQUI
0068          00800      END     ;

```

#### 4.1.5. Inserción de un mensaje en la pantalla: MES

Los dibujos representativos de los caracteres ASCII utilizados por el Basic se guardan dentro de una tabla en memoria muerta. Esta tabla se sitúa entre las direcciones 3D00H y 3FFFH, ambas inclusive. Contiene todos los códigos ASCII a partir del código de número 32 (espacio), hasta el código de número 127 (©), en el orden de los códigos crecientes. Cada dibujo es memorizado por los ocho octetos que constituyen el carácter.

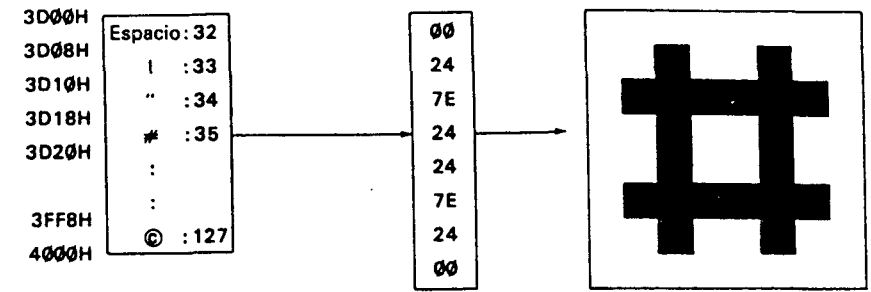


Tabla de los dibujos de los caracteres ASCII

Octetos del carácter #

Dibujo del carácter #

El subprograma MES utiliza esta tabla para dibujar los caracteres ASCII en la pantalla. MES utiliza el octeto almacenado en la variable ATRB como atributo para el dibujo de los caracteres de un mensaje. Este se da bajo la forma de una serie de caracteres ASCII (obtenida, por ejemplo, con la orden DEFM), pudiendo contener algunos caracteres especiales (obtenidos mediante las órdenes DEFB o DEFW). El subprograma MES reconoce cuatro tipos de caracteres especiales que permiten actuar sobre el desarrollo de la visualización del mensaje:

Carácter especial	Significado
0	Este carácter indica el fin del mensaje.
13	Provoca un salto de línea (retorno de carro).
2 seguido de un atributo	Coloca el octeto siguiente del mensaje en la variable ATRB que contiene el atributo para el dibujo de los caracteres.
127 + n	Salto de n espacios (1 ≤ n ≤ 128).

He aquí un ejemplo de mensaje que puede ser utilizado por MES:

```

0000 BC      00100 MESSA  DEFB   140      ;DECALAJE PARA CENTRADO
0001 02      00110      DEFB   2,31H   ;AZUL SOBRE FONDO AMARILLO
0003 42      00120      DEFM   'Bonjour'
0004 6F 6E 6A 6F 76 72  DEFB   13,13   ; SALTOS DE LINEA
000A 0D      00130      DEFB   2,0B6H ; NEGRO SOBRE BLANCO Y
000C 02      00140      DEFB   B8      PARRADEFEO
000E 4A      00150      DEFM   'Je suis le ZX SPECTRUM'
0010 66 20 73 76 69 73 20 6C
0011 66 20 6A 68 20 63 60 46
0012 43 64 62 65 4D
0024 00      00160      DEFB   0       ; FIN DEL MENSAJE

```

Listado del programa MES:

```

00010 .....
00020 .....
00030 SUBPROGRAMA DE VISUALIZACION DE UN MENSAJE EN LA PANTALLA
00040 .....
00050 ENTRADA: D = NUMERO DE LINEA DONDE COMIENZA LA VISUALIZACION
00060 E = NUMERO DE COLUMNA DONDE COMIENZA LA VISUALIZACION
00070 HL = DIRECCION DEL PRIMER OCTETO DEL MENSAJE
00080 .....
00090 .....
00100 .....
0000 7E 00110 MES LD A,(HL) ; CARACTER EN A
0001 23 00120 INC HL ; INCREMENTAR PUNTERO
0002 B7 00130 OR A ; ¿COMPROBAR SI A = 0?
0003 C8 00140 RET Z ; SI ES SI RETORNO; FIN DEL MENSAJE
0004 FE0D 00150 CP ODH ; ¿COMPROBAR A = RETORNO DE CARRO?
0006 2005 00160 JR NZ,ZI ; NO
0008 14 00170 INC D ; BALTAR UNA LINEA
0009 1E00 00180 LD E,O ; COLOCARSE DE NUEVO AL INICIO DE LA LINEA
000B 18F3 00190 JR MES ; PASAR AL CARACTER SIGUIENTE
000D CB7F 00200 ZI BIT 7,A ; ¿COMPROBAR SI A >= 128?
000F 2811 00210 JR Z,ZJ ; NO
0011 D67F 00220 SUB 127 ; A CONTIENE EL NUMERO DE ESPACIOS
0013 83 00230 ADD A,E ; AÑADIR AL NUMERO DE COLUMNAS
0014 4F 00240 LD C,A ; GUARDAR RESULTADO EN C
0018 E61F 00250 AND 1FH ; MASCARA PARA RECUPERAR EL NUMERO
0017 EF 00260 LD E,A ; DE COLUMNAS
0018 79 00270 LD A,C ; RECUPERAR RESULTADO
0019 E6E0 00280 AND OEOH ; NUMERO DE LINEAS * 32
001B 07 00290 RLCA ; DIVIDIR POR 32 PARA OBTENER
001C 07 00300 RLCA ; EL NUMERO DE LINEAS
001D 07 00310 RLCA
001E 82 00320 ADD A,D ; AÑADIR RESULTADO A D
001F 87 00330 LD D,A ; NUEVO NUMERO DE LA LINEA
0020 18DE 00340 JR MES ; PASAR AL SIGUIENTE CARACTER
0022 FE02 00350 ZJ CP 2 ; ¿COMPROBAR SI A = 2?
0024 2007 00360 JR NZ,ZK ; NO
0026 7E 00370 LD A,(HL) ; LECTURA DE ATRIBUTO
0027 23 00380 INC HL ; INCREMENTAR PUNTERO
0028 328600 00390 LD (ATRB),A ; COLOCAR EL ATRIBUTO EN ATRB
002B 18D3 00400 JR MES ; PASAR AL SIGUIENTE CARACTER
002D CD3A00 00410 ZK CALL AFI ; VISUALIZACION DEL CARACTER ASCII
0030 1C 00420 INC E ; COLUMNA SIGUIENTE
0031 CB6B 00430 BIT 5,E ; ¿COLUMNA > = 32?
0033 28CB 00440 JR Z,MES ; NO
0038 14 00450 INC D ; PASAR A LA LINEA SIGUIENTE
0036 1E00 00460 LD E,O ; COLOCARSE AL INICIO DE LA LINEA
0038 18C8 00470 JR MES ; PASAR AL SIGUIENTE CARACTER
00480 .....
00490 .....
00500 SUBPROGRAMA DE VISUALIZACION DE UN CARACTER ASCII
00510 .....
00520 ENTRADA: D = NUMERO DE LINEA
00530 E = NUMERO DE COLUMNA
00540 A = CODIGO ASCII DEL CARACTER
00550 .....
00560 NO SE MODIFICA NINGUN REGISTRO SALVO AF
00570 .....
00580 .....
00590 .....
003A C8 00600 AFI PUSH BC ; GUARDAR REGISTROS
003B E8 00610 PUSH HL ;
003C D6 00620 PUSH DE ;
003D F8 00630 PUSH AF ;
003E CD8F00 00640 CALL TRAN ; BUSQUEDA DIRECCION DEL PUNTO
0041 F1 00650 POP AF ; CARACTER ASCII
0042 2800 00660 LD H,O ;
0044 6F 00670 LD LA ; CODIGO CARACTER EN HL
0046 29 00680 ADD HL,HL ; CODIGO . 2
0048 29 00690 ADD HL,HL ; CODIGO . 4
0047 29 00700 ADD HL,HL ; CODIGO . 8
0048 01003C 00710 LD BC,3COOH ; INICIO ZONA DE DIBUJO DE LOS CARACTERES -256
004B 09 00720 ADD HL,BC ; DIRECCION DEL DIBUJO DEL CARACTER
004C 0608 00730 LD B,S ; 8 OCTETOS EN EL CARACTER

```

```

004E 7E 00740 ZM LD A,(HL) ; OCTETO DE LA TABLA
004F 12 00760 LD (DE),A ; TRANSFERENCIA A LA MEMORIA VIDEO
0050 23 00780 INC HL ; OCTETO SIGUIENTE DE LA TABLA
0051 14 00770 INC D ; DIRECCION VIDEO SIGUIENTE
0052 10FA 00780 DJNZ ZM ; BUCLE PARA TRANSFERIR LOS 8 OCTETOS
0054 D1 00790 POP DE ; POSICION EN LA PANTALLA
0056 CD7800 00800 CALL TATR ; BUSQUEDA DE LA DIRECCION DEL OCTETO DE
0058 3A8600 00810 LD A,(ATRB) ATRIBUTOS
005B 77 00820 LD (HL),A ; ATRIBUTO A UTILIZAR
005C E1 00830 POP HL ; ASIGNAR ESTE ATRIBUTO AL CARACTER
005D C1 00840 POP BC ; RESTAURAR REGISTROS
005E C9 00850 RET ;

```

```

00860 .....
00870 SUBPROGRAMA DE CONVERSION
00880 .....
00890 ENTRADA: D = NUMERO DE LINEA
00900 E = NUMERO DE COLUMNA
00910 .....
00920 SALIDA: DE = DIRECCION DEL PRIMER OCTETO DEL CARACTER
00930 EN MEMORIA VIDEO DE LOS PUNTOS
00940 .....
00950 .....
00960 .....
00970 .....

```

```

005F 7A 00980 TRAN LD A,D ; NUMERO DE COLUMNA
0060 1840 00990 LD D,40H ; INICIO PRIMERA ZONA
0062 D608 01000 SUB 8 ; QUITAR LONGITUD DE LA ZONA
0064 380A 01010 JR C,ZL ; SI PRIMERA ZONA
0066 1648 01020 LD D,48H ; INICIO SEGUNDA ZONA
0068 D608 01030 SUB 8 ; QUITAR LONGITUD DE LA ZONA
006A 3804 01040 JR C,ZL ; SI SEGUNDA ZONA
006C 1650 01050 LD D,50H ; ULTIMA ZONA
006E D608 01060 SUB 8 ; QUITAR LONGITUD DE LA ZONA
0070 C808 01070 ZL ADD A,S ; NUMERO DE LINEA EN LA ZONA
0072 0F 01080 RRC A ; MULTIPLICACION DEL NUMERO DE
0073 0F 01090 RRC A ; LINEA POR 32
0074 0F 01100 RRC A ;
0076 83 01110 ADD A,E ; AÑADIR A COLUMNA PARA OBTENER
0078 8F 01120 LD E,A ; EL OCTETO DE MENOS PESO DE
0077 C9 01130 RET ; LA DIRECCION VIDEO

```

```

01140 .....
01150 SUBPROGRAMA DE CONVERSION
01160 .....
01170 ENTRADA: D = NUMERO DE LINEA
01180 E = NUMERO DE COLUMNA
01190 .....
01200 SALIDA: HL = DIRECCION DEL OCTETO DE ATRIBUTOS ASOCIADO
01210 .....
01220 .....
01230 .....
01240 .....

```

```

0078 2800 01250 TATR LD H,O ;
007A 6A 01260 LD L,D ; HL CONTIENE EL NUMERO DE LINEA
007B 29 01270 ADD HL,HL ; NUMERO DE LINEA . 2
007C 29 01280 ADD HL,HL ; NUMERO DE LINEA . 4
007D 29 01290 ADD HL,HL ; NUMERO DE LINEA . 8
007E 29 01300 ADD HL,HL ; NUMERO DE LINEA . 16
007F 29 01310 ADD HL,HL ; NUMERO DE LINEA . 32
0080 D5 01320 PUSH DE ; GUARDAR POSICION
0081 1858 01330 LD D,58H ; INICIO ZONA DE ATRIBUTOS
0083 19 01340 ADD HL,DE ; HL = DIRECCION OCTETO DE ATRIBUTOS
0084 D1 01350 POP DE ; RESTAURAR POSICION
0085 C9 01360 RET ;
0001 01370 ATRB DEFB 1 ; ATRIBUTO A UTILIZAR
0000 01380 END ;

```

#### 4.1.6. Visualización de caracteres gráficos

Es fácil modificar el programa MES precedente para que permita la visualización de caracteres gráficos. Para ello hagamos que el carácter especial 1 solicite la visualización del carácter gráfico cuyos ocho caracteres se encuentran después del carácter especial 1.

El mensaje gráfico siguiente podrá ser utilizado con el subprograma MES modificado:

```
MESA  DEFB      2, 2      ; Rojo sobre fondo negro
      DEFB      1        ; Carácter gráfico
      DEFB      14H, 1CH, 14H, 5DH, 7EH, 5DH, 77H
      DEFB      13       ; Salto de línea
      DEFM      'Bonjour'
      DEFB      Ø
```

He aquí el listado del subprograma MES modificado al que habrá que añadir los tres subprogramas (AFI, TRAN, TATR) que MES utiliza.

```
00010 : .....
00020 : .....
00030 : SUBPROGRAMA DE VISUALIZACION DE UN MENSAJE EN LA PANTALLA
00040 : .....
00050 : ENTRADA: D = NUMERO DE LINEA DONDE COMIENZA LA VISUALIZACION
00060 : E = NUMERO DE COLUMNA DONDE COMIENZA LA VISUALIZACION
00070 : HL = DIRECCION DEL PRIMER OCTETO DEL MENSAJE
00080 : .....
00090 : .....
00100 : .....
00110 MESA LD A,(HL) ; CARACTER EN A
00120 INC HL ; INCREMENTAR PUNTERO
00130 OR A ; ¿COMPROBAR SI A = Ø?
00140 RET Z ; SI ES SI RETORNO; FIN DEL MENSAJE
00150 CP ODH ; ¿COMPROBAR SI A = RETORNO DE CARRO?
00160 JR NZ,ZI ; NO
00170 INC D ; SALTAR UNA LINEA
00180 LD E,Ø ; COLOCARSE DE NUEVO EN INICIO DE LINEA
00190 JR MES ; PASAR AL CARACTER SIGUIENTE
00200 ZI BIT 7,A ; ¿COMPROBAR SI A > = 128?
00210 JR Z,ZJ ; NO
00220 SUB 127 ; A CONTIENE EL NUMERO DE ESPACIOS
00230 ADD A,E ; AÑADIR AL NUMERO DE COLUMNAS
00240 LD C,A ; GUARDAR RESULTADO EN C
00250 AND 1FH ; MASCARA PARA RECUPERAR EL NUMERO
00260 LD E,A ; DE COLUMNAS
00270 LD A,C ; RECUPERAR RESULTADO
00280 AND 0E0H ; NUMERO DE LINEAS . 32
00290 RLCA ; DIVIDIR POR 32 PARA OBTENER
00300 RLCA ; EL NUMERO DE LINEAS
00310 RLCA
00320 ADD A,D ; AÑADIR RESULTADO A D
00330 LD D,A ; NUEVO NUMERO DE LINEA
00340 JR MES ; PASAR AL CARACTER SIGUIENTE
00350 ZJ CP 1 ; ¿COMPROBAR SI A = 1?
00360 JR NZ,ZP ; NO
00370 PUSH DE ; GUARDAR POSICION
00380 CALL TRAN ; BUSQUEDA DE LA DIRECCION DE LA MEMORIA VIDEO
00390 LD B,B ; 8 OCTETOS POR CARACTER
00400 ZQ LD A,(HL) ; OCTETO DEL CARACTER
00410 LD (DE),A ; VISUALIZACION
00420 INC HL ; INCREMENTAR PUNTERO
00430 INC D ; OCTETO SIGUIENTE DEL CARACTER
```

```
0030 10FA 00440 DJNZ ZQ ; BUCLE PARA VISUALIZAR LOS 8 OCTETOS
0032 D1 00450 POP DE ; RESTAURAR POSICION
0033 E5 00460 PUSH HL ; GUARDAR PUNTERO
0034 CD9400 00470 CALL TATR ; BUSQUEDA DIRECCION DEL OCTETO DE ATRIBUTOS
0037 3AA200 00480 LD A,(ATRB) ; ATRIBUTO A UTILIZAR
003A 77 00490 LD (HL),A ; ASIGNAR ESTE ATRIBUTO AL CARACTER
003B E1 00800 POP HL ; RESTAURAR PUNTERO
003C 180E 00610 JR ZR ; PASAR AL CARACTER SIGUIENTE
003E FE02 00620 CP 2 ; ¿COMPROBAR SI A = 2?
0040 2007 00630 JR NZ,ZK ; NO
0042 7E 00840 LD A,(HL) ; LECTURA ATRIBUTO
0043 25 00650 INC HL ; INCREMENTAR PUNTERO
0044 32A200 00660 LD (ATRB),A ; COLOCAR EL ATRIBUTO EN ATRB
0047 18B7 00570 JR MES ; PASAR AL CARACTER SIGUIENTE
0049 CD5600 00680 ZK CALL AFI ; VISUALIZACION DEL CARACTER ASCII
004C 1C 00690 ZR INC E ; COLUMNA SIGUIENTE
004D CB6B 00600 BIT 5,E ; ¿COLUMNA > = 32?
004F 28AF 00610 JR Z,MES ; NO
0051 14 00620 INC D ; PASAR A LA LINEA SIGUIENTE
0052 1E00 00630 LD E,Ø ; COLOCARSE EN INICIO DE LINEA
0054 18AA 00640 JR MES ; PASAR AL CARACTER SIGUIENTE
```

#### 4.2. La impresora

La impresora del SPECTRUM tiene una resolución idéntica a la de la pantalla, lo que permite imprimir tanto textos (caracteres ASCII) como dibujos (caracteres gráficos). Está concebida para visualizar de una sola vez una línea de 32 caracteres. Así, el interpretador Basic controla un *buffer* de impresora de 256 octetos destinado a contener los 32 caracteres que hay que enviar hacia la impresora. Este *buffer* se sitúa entre las direcciones 5B00H y 5BFFH. Los 32 primeros octetos del *buffer* contendrán los primeros 32 caracteres, los 32 octetos siguientes contendrán los segundos octetos de los 32 caracteres y así sucesivamente hasta los octavos octetos. El interpretador Basic controla una variable situada en la dirección 2368H que contiene el número de caracteres presentes en el *buffer*. Cuando el *buffer* está lleno, o cuando el carácter «retorno del carro» (código 13) es enviado a la impresora, el interpretador Basic envía el contenido del *buffer* hacia la impresora para sacarlo sobre papel, gracias al subprograma situado en la dirección ECDH. Si la impresora no está ligada al SPECTRUM, este subprograma no tiene ningún efecto.

El subprograma PRINT que indicamos seguidamente utiliza el subprograma situado en la dirección ECDH y la variable situada en la dirección 2368H para visualizar un mensaje constituido por caracteres ASCII, por retornos de carro y terminando por un octeto nulo.

Ejemplo de mensaje:

```
MESSA  DEFM      «Hola»
      DEFB      13
      DEFM      «Yo soy el SPECTRUM»
      DEFB      13,Ø
```

Dado que la impresión sólo es ordenada si el buffer está lleno o si se envía un retorno de carro, deberá terminarse el mensaje por un retorno de carro, salvo si su medida es múltiplo de 32.

```

OECD      00010 IMP EQU OECDH ; SUBPROGRAMA DE IMPRESION DEL
00020      ; BUFFER DE IMPRESORA
BC80      00030 NCAR EQU 23680 ; NUMERO DE CARACTERES EN EL BUFFER
.....
00060      SUBPROGRAMA DE VISUALIZACION DE UN MENSAJE EN LA IMPRESORA
00070      ENTRADA: HL = DIRECCION DEL PRIMER OCTETO DEL MENSAJE
00080      .....
00110
0000 7E   00120 PRINT LD A,(HL) ; CARACTER EN A
0001 23   00130 INC HL ; INCREMENTAR PUNTERO
0002 B7   00140 OR A ; ¿COMPROBAR SI A = 0?
0003 C8   00150 RET Z ; SI ES SI RETORNO, FIN DEL MENSAJE
0004 E8   00160 PUSH HL ; GUARDAR PUNTERO
0006 FE0D 00170 CP ODH ; ¿COMPROBAR SI A = RETORNO DE CARRO?
0007 2824 00180 JR Z,ZS ; SI
0009 2600 00190 LD H,O
000B 6F   00200 LD LA ; CODIGO DEL CARACTER EN HL
000C 29   00210 ADD HL,HL ; CODIGO . 2
000D 29   00220 ADD HL,HL ; CODIGO . 4
000E 29   00230 ADD HL,HL ; CODIGO . 8
000F 01003C 00240 LD BC,300H; INICIO ZONA DEL DIBUJO DE LOS CARACTERES - 256
0012 09   00250 ADD HL,BC ; DIRECCION DEL DIBUJO DEL CARACTER
0013 3A808C 00260 LD A,(NCAR); PUNTERO SOBRE EL BUFFER DE IMPRESORA
0018 8F   00270 LD EA
0017 165B 00280 LD D,BBH ; DE = PUNTERO SOBRE EL BUFFER DE IMPRESORA
0019 0608 00290 LD B,B ; 8 OCTETOS EN EL CARACTER
001B 7E   00300 ZT LD A,(HL) ; OCTETO DE LA TABLA
001C 12   00310 LD (DE),A ; TRANSFERIDO A LA MEMORIA VIDEO
001D 23   00320 INC HL ; OCTETO SIGUIENTE DE LA TABLA
001E 7B   00330 LD A,E ; PASAR AL OCTETO SIGUIENTE DEL
001F 6820 00340 ADD A,2 ; CARACTER EN EL BUFFER
0021 8F   00350 LD EA ; DE IMPRESORA
0022 10F7 00360 DJNZ ZT ; BUCLE PARA TRANSFERIR LOS 8 OCTETOS
0024 3A808C 00370 LD A,(NCAR);
0027 3C   00380 INC A ; INCREMENTAR NUMERO DE CARACTERES
0028 32808C 00390 LD (NCAR),A; EN EL BUFFER DE IMPRESORA
002B FE20 00400 CP 32 ; ¿FIN DE LINEA?
002D 0C0D0E 00410 ZS CALL Z,IMP ; SI ES SI IMPRESION DE LINEA
0030 E1   00420 POP HL ; RESTAURAR PUNTERO
0031 18CD 00430 JR PRINT ; PASAR AL CARACTER SIGUIENTE
0000      00440 END

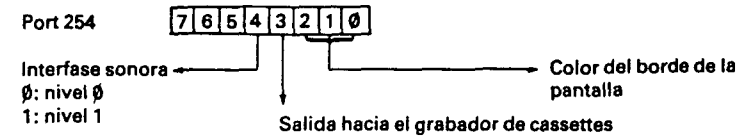
```

Del mismo modo que para el subprograma MES, PRINT podrá ser modificado para permitir la impresión de caracteres gráficos.

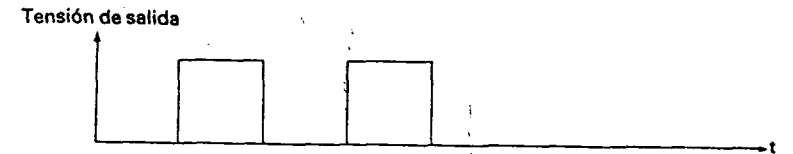
### 4.3. La interfase sonora

La interfase sonora del ZX SPECTRUM consta de una salida programable que puede tomar dos niveles diferentes y que está ligada a un altavoz a través de un pequeño amplificador.

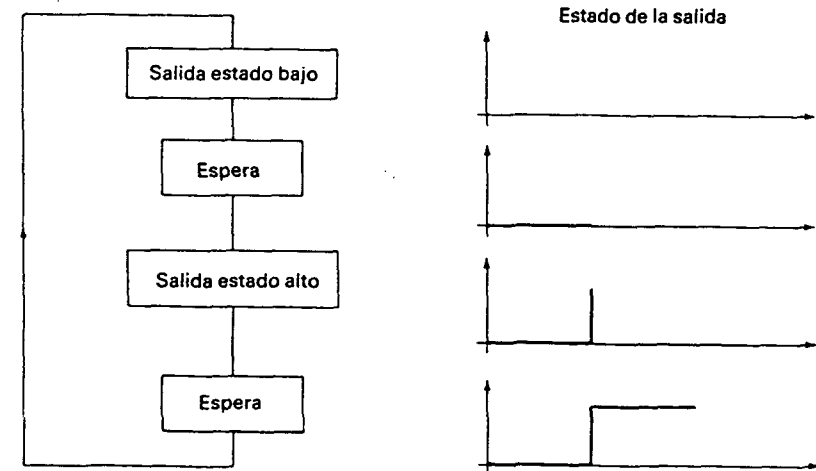
La programación de esta salida se realiza mediante el bit 4 del port 254. Este port sirve igualmente para la programación del borde de la pantalla y para la escritura de informaciones en el cassette.



Haciendo variar rápidamente la tensión tomada por esta salida, se genera una señal eléctrica que producirá un sonido al estar ligada a un amplificador con un altavoz. De este modo es posible crear una señal cuadrada cuya frecuencia podrá ajustarse con el fin de producir la nota musical deseada. Esta señal se representa por medio del esquema siguiente:



Para generar esta señal se utilizará un programa que responda al diagrama del flujo siguiente:



Para que el sonido no dure eternamente como en este diagrama de flujo, tendrá que utilizarse un segundo contador que limite el número de pulsos generados. Así será posible modificar la duración de la nota, escogiendo el número de pulsos enviados, y modificar su tono, disminuyendo o aumentando la duración creada por el bucle de espera. Cuanto más grande sea el bucle de espera más grave será el sonido, y cuanto más pequeño, más agudo.

```

00010 .....
00020 .....
00030 EL PROGRAMA SIGUIENTE GENERA UNA NOTA DE DURACION
00040 LIMITADA Y PARA EL MICROPROCESADOR (INSTRUCCION HALT)
00050 .....
00060 .....
00070 .....
8000 00080 ORG 8000H
0007 00090 COLOR EQU 7 ; COLOR BLANCO PARA EL BORDE
8000 F3 00100 INICIO DI ; PROHIBIR INTERRUPCIONES PARA
00110 ; TENER UN SONIDO PURO
8001 3E07 00120 LD A,COLOR ; COLOR DEL BORDE EN A
8003 0E00 00130 LD C,0 ; CONTADOR DE DURACION DE LA NOTA
8005 D3FE 00140 LOOP OUT (284),A ; MODIFICACION DE LA SALIDA
8007 06C8 00150 LD B,200 ; CONTADOR DE BUCLE DE ESPERA
8009 10FE 00160 WAIT DJNZ WAIT ; ESPERA
800B EE10 00170 XOR 10H ; EL BIT 4 DE A ES COMPLEMENTADO
800D 0D 00180 DEC C ; DECREMENTAR CONTADOR DE DURACION
800E 20F6 00190 JR NZ,LOOP ; SI EL CONTADOR NO ES NULO CONTINUAR
8010 78 00200 HALT ; PARO DEL Z 80
8000 00210 END INICIO

```

En el programa de ensamblador siguiente es posible evaluar la frecuencia de la nota generada calculando la duración producida por el bucle de espera. Entre dos valores de la tensión de salida, el microprocesador utiliza un número de ciclos de reloj igual a  $11 + 7 + 149 + 13 + 8 + 7 + 12 = 1986$ . Si la frecuencia del reloj del ZX SPECTRUM está fijada a 3,25 MHz, la del sonido generado será 3,25 MHz/1986, o sea 1636 Hz (frecuencia audible).

Las notas musicales tienen una frecuencia determinada. También es posible calcular el número de inicialización del contador de bucle de espera para producir la nota deseada. A este número le llamaremos N. La frecuencia generada viene dada por la fórmula siguiente:

$$F = \frac{3,25 \cdot 10^6}{46 + (N - 1) \cdot 3} \text{ de donde } N = \frac{1}{13} \left( \frac{3,25 \cdot 10^6}{F} - 46 \right) + 1$$

Dado que la duración del bucle de espera es variable, según la nota emitida deberá ajustarse el contador de duración para generar notas de la misma duración. Sean D<sub>1</sub> o D<sub>2</sub> los valores que inicializan a los contadores de bucle de duración respectivamente para las frecuencias F<sub>1</sub> y F<sub>2</sub>. Tenemos la siguiente fórmula:

$$\frac{D_1}{F_1} = \frac{D_2}{F_2}$$

Una vez escogida la duración de una nota, esta fórmula permitirá determinar todos los números D que sirven para inicializar los contadores de bucle. Los valores de estos parámetros correspondientes a la escala se dan en la siguiente tabla:

Nota	DO	RE	MI	FA	SOL	LA	SI	DO
Frecuencia (Hz)	1047	1175	1319	1397	1568	1760	1976	2093
Contador de bucle de espera	236	210	187	176	157	140	124	117
Contador de bucle de duración	128	143	161	170	191	214	241	255

El siguiente programa recorre constantemente la escala en un sentido y luego en el otro. Con el fin de animar un juego podrá utilizarse este programa para generar una pequeña melodía al final de la partida, modificando los valores de las constantes reagrupadas en la tabla TAB.

```

00010 .....
00020 .....
00030 EL SIGUIENTE PROGRAMA GENERA ETERNAMENTE LA
00040 ESCALA EN UN SENTIDO Y EN OTRO
00050 .....
00060 .....
00070 .....
8000 00080 ORG 8000H
0007 00090 COLOR EQU 7 ; COLOR BLANCO PARA EL BORDE
8000 F3 00100 INICIO DI ; PROHIBIR INTERRUPCIONES PARA
00110 ; TENER UN SONIDO PURO
8001 213380 00120 LD HL,TAB ; INICIALIZACION DEL PUNTERO SOBRE TAB
00130 ;
00140 ; RECORRER ESCALA AL DERECHO
00150 ;
8004 1E08 00160 BUCLE LD E,8 ; 8 NOTAS EN LA ESCALA
8006 4E 00170 U LD C,(HL) ; CONTADOR DE BUCLE DE ESPERA
8007 23 00180 INC HL ; INCREMENTAR PUNTERO
8008 58 00190 LD D,(HL) ; CONTADOR DE DURACION
8009 23 00200 INC HL ; INCREMENTAR PUNTERO
800A CD1E80 00210 CALL SON ; GENERACION DE UNA NOTA
800D 1D 00220 DEC E ; CONTADOR DE NUMERO DE NOTAS
800E 20F6 00230 JR NZ,ZU ; PASAR A LA NOTA SIGUIENTE
00240 ;
00250 ; RECORRER LA ESCALA A LA INVERSA
00260 ;
8010 1E08 00270 LD E,8 ; 8 NOTAS EN LA ESCALA
8012 2B 00280 ZV DEC HL ; DECREMENTAR EL PUNTERO
8013 56 00290 LD D,(HL) ; CONTADOR DE DURACION
8014 2B 00300 DEC HL ; DECREMENTAR PUNTERO
8015 4E 00310 LD C,(HL) ; CONTADOR DE BUCLE DE ESPERA
8016 CD1E80 00320 CALL SON ; GENERACION DE UNA NOTA
8019 1D 00330 DEC E ; CONTADOR DE NUMERO DE NOTAS
801A 20F6 00340 JR NZ,ZV ; PASAR A LA NOTA SIGUIENTE
801C 18E6 00350 JR BUCLE ; RECOMENZAR ETERNAMENTE

```



```

00360 .....
00370 .....
00380 SUEPROGRAMA DE SONIDO
00390 .....
00400 ENTRADA: D = CONTADOR DE BUCLE DE DURACION
00410 C = CONTADOR DE BUCLE DE ESPERA
00420 .....
00430 .....
00440 .....
801E 3E07 00460 SON LD A,COLOR ; COLOR DEL BORDE EN A
8020 D3FE 00460 LOOP OUT (284),A ; MODIFICACION DE LA SALIDA
8022 41 00470 LD B,C ; CONTADOR DE BUCLE DE ESPERA
8023 10FE 00480 WAIT DJNZ WAIT ; ESPERA
8028 EE10 00490 XOR 10H ; EL BIT 4 DE A ES COMPLEMENTADO
8027 18 00500 DEC D ; DECREMENTAR EL CONTADOR DE DURACION
8028 20F8 00810 JR NZ,LOOP ; SI EL CONTADOR NO ES NULO CONTINUAR
802A 010004 00820 LD BC,400H ; CONTADOR DE BUCLE DE ESPERA
802D 0B 00830 LA DEC BC ; ESPERA ENTRE CADA NOTA
802E 78 00840 LD A,B ;
802F B1 00850 OR C ; ¿COMPROBAR SI BC = 0?
8030 20FB 00860 JR NZ,LA ; SI BC < > 0 BUCLE
8032C9 00870 RET ; RETORNO
00880 .....
00890 TABLA DE LAS NOTAS
00900 .....
00900 .....
8033 EC 00610 TAB DEFB 236 ; ESPERA DO
8034 80 00620 DEFB 128 ; DURACION DO
8035 D2 00630 DEFB 210 ; ESPERA RE
8036 8F 00640 DEFB 143 ; DURACION RE
8037 BB 00650 DEFB 187 ; ESPERA MI
8038 A1 00660 DEFB 161 ; DURACION MI
8039 B0 00670 DEFB 176 ; ESPERA FA
803A AA 00680 DEFB 170 ; DURACION FA
803B 9D 00690 DEFB 187 ; ESPERA SOL
803C BF 00700 DEFB 191 ; DURACION SOL
803D 8C 00710 DEFB 140 ; ESPERA LA
803E D6 00720 DEFB 214 ; DURACION LA
803F 7C 00730 DEFB 124 ; ESPERA SI
8040 F1 00740 DEFB 241 ; DURACION SI
8041 76 00750 DEFB 117 ; ESPERA DO
8042 FF 00760 DEFB 255 ; DURACION DO
8000 00770 END INICIO

```

Para obtener sonoridades más complejas, la duración producida por el bucle de espera deberá modularse, modificando su duración a lo largo del tiempo. La modificación más simple consiste en decrementar la variable que inicializa el contador del bucle de espera cada vez que la salida cambie de estado. De esta manera se obtiene un sonido muy interesante que se utiliza en numerosos juegos.

```

00010 .....
00020 .....
00030 EL SIGUIENTE PROGRAMA GENERA UN SONIDO DE DURACION
00040 LIMITADA Y PARA EL MICROPROCESADOR (INSTRUCCION HALT)
00050 .....
00060 .....
00070 .....
8000 00080 ORG 8000H
0007 00090 COLOR EQU 7 ; COLOR BLANCO PARA EL BORDE
8000 F3 00100 INICIO DI ; PROHIBIR INTERRUPCIONES PARA
; TENER UN SONIDO PURO
00110 .....
8001 3E07 00120 LD A,COLOR ; COLOR DEL BORDE EN A
8003 0E00 00130 LD C,0 ; CONTADOR DE DURACION DE SONIDO
8006 D3FE 00140 LOOP OUT (284),A ; MODIFICACION DE LA SALIDA
8007 41 00150 LD B,C ; CONTADOR DE BUCLE DE ESPERA
8008 10FE 00160 WAIT DJNZ WAIT ; ESPERA
800A EE10 00170 XOR 10H ; EL BIT 4 DE A ES COMPLEMENTADO
800C 0D 00180 DEC C ; DECREMENTAR CONTADOR DE DURACION
800D 20F8 00190 JR NZ,LOOP ; SI EL CONTADOR NO ES NULO CONTINUAR
800F 78 00200 HALT ; PARO DEL Z 80
8000 00210 END INICIO

```

Modificando la duración del bucle de espera por procedimientos diferentes se obtendrán sonoridades más o menos buenas. Experimentélas y retenga aquellas que le parezcan mejores. Combinando estas sonoridades elementales, a veces podrán obtenerse otras mejores.

#### 4.4. El teclado

Ocho ports de entrada/salida permiten leer todas las teclas del teclado. Cada port permite leer una media hilera de cinco teclas tal como se indica en la tabla siguiente:

Número del port	Media hilera	bit 0	bit 1	bit 2	bit 3	bit 4
FEFEH	CAPS SHIFT à V	CAPS SHIFT	Z	X	C	V
FDFEH	A à G	A	S	D	F	G
FBFEH	Q à T	Q	W	E	R	T
F7FEH	1 à 5	1	2	3	4	5
EFFEH	0 à 6	0	9	8	7	6
DFFEH	P à Y	P	O	I	U	Y
BFFEH	ENTER à H	ENTER	L	K	J	H
7FFEH	SPACE à B	SPACE	SYMBOL SHIFT	M	N	B

Una tecla está pulsada si el bit correspondiente del port es nulo. Si este bit vale 1, la tecla no está pulsada.

Contrariamente al port 254 utilizado para la interfase sonora, y para especificar el borde de la pantalla que era un port de ocho bits, los ports que se indican a continuación son ports de 16 bits. Será necesario, pues, especificar el octeto de más peso y el octeto de menos peso del port al realizar la lectura de uno de esos ports.

Ejemplo:

```

LD A, 7FH ; octeto de más peso del port 7FFEh
IN A, (0FEH) ; lectura del port número 7FFEh
BIT 0, A ; compruebe si la tecla SPACE está pulsada
JR Z, BREAK ; sí, ir a BREAK

```

Cuando las interrupciones están autorizadas, el subprograma de interrupción situado en la dirección 38H efectúa una lectura del teclado cada 20 milisegundos. Si una tecla está pulsada evalúa el código ASCII de esta tecla y la coloca en la posición de memoria situada en la dirección 23560. A continuación coloca a 1 el bit 5 del octeto situado en la dirección 23611 para indicar que se ha pulsado una tecla. Es este subprograma el que controla la temporización del teclado y la repetición automática de teclas.

Si se prohíben las interrupciones en un programa en lenguaje máquina y volvemos al monitor Basic, este último no aceptará ningún carácter pulsado en el teclado y no tendremos otra solución que desconectar el SPECTRUM. Debemos pensar en esto al escribir subprogramas en lenguaje máquina que prohíban las interrupciones.

Cuando éstas están autorizadas podremos utilizar las informaciones suministradas por el subprograma de interrupción en un programa en ensamblador para obtener el código ASCII de la última tecla pulsada. Esto lo realiza el siguiente subprograma:

```

00010 .....
00020 .....
00030 SUBPROGRAMA DE LECTURA DEL TECLADO
00040 .....
00080 SALIDA: Z = VALE 1 SI NINGUNA TECLA HA SIDO PULSADA
00060 Z = VALE 0 SI UNA TECLA HA SIDO PULSADA
00070 Y A CONTIENE EL CODIGO ASCII DE LA TECLA
00080 .....
00080 NINGUN REGISTRO ES MODIFICADO SALVO AF
00100 .....
00110 .....
00120 .....
0000 3A3B8C 00130 KEY LD A,(23611) ;INDICADORES
0003 CB8F 00140 BIT 5,A ;¿TECLA PULSADA?
0006 C8 00180 RET Z ;NO RETORNO CON Z = 1
0006 CBAF 00160 RES 8,A ;COLOCAR DE NUEVO EL INDICADOR A CERO
0006 323B8C 00170 LD (23611),A
000B 3A085C 00180 LD A,(23560) ;CODIGO ASCII DE LA TECLA PULSADA
000E C9 00190 RET ;RETORNO CON Z = 0
0000 00200 END

```

Cuando deseemos conocer instantáneamente el estado del teclado deberemos leer las teclas pulsadas leyendo directamente los ports correspondientes. Es lo que empleamos en los juegos de acción rápida. Contrariamente, cuando queramos leer el teclado para conocer cierto número de caracteres ASCII, deberemos utilizar las informaciones suministradas por el subprograma de interrupción que controla la temporización del teclado. Recordemos que en este caso no deberemos modificar el registro IY utilizado en las interrupciones.

El subprograma siguiente emplea el subprograma KEY para leer una serie de caracteres ASCII como lo hace el interpretador Basic. La tecla < DELETE > (< CAPS SHIFT > + < Ø >), sirve para borrar el último carácter pulsado y la tecla < ENTER > indica el fin de la línea. Este subprograma, además, utiliza los subprogramas AFI, TRAN, y TATR indicados en el párrafo 4.1.5.

```

00010 .....
00020 .....
00030 SUBPROGRAMA DE LECTURA DE UNA SERIE DE CARACTERES
00040 ASCII PULSADOS SOBRE EL TECLADO
00080 .....
00060 ENTRADA: D = NUMERO DE LINEA DONDE COMIENZA LA ENTRADA
00070 E = NUMERO DE COLUMNA DONDE COMIENZA LA ENTRADA
00080 C = NUMERO MAXIMO DE CARACTERES A LEER
00090 HL = DIRECCION DEL INICIO DE LA ZONA DE MEMORIA
00100 DESTINADA A RECIBIR LA LINEA LEIDA
00110 .....
00120 SALIDA: B = NUMERO DE CARACTERES LEIDOS
00130 .....
00140 .....
00180 .....
0000 0600 00160 CUR LD B,0 ;NINGUN CARACTER LEIDO
0002 CD4A00 00170 ZY CALL KEY ;LECTURA DE CARACTER
0005 281A 00180 JR Z,ZW ;SI NO HAY TECLA PULSADA
0007 FE0C 00190 CP 12 ;¿ES ESTA LA TECLA DELETE?
0009 282A 00200 JR Z,ZX ;SI ES SI, TRATAR ESTE CASO
000B FE0D 00210 CP 13 ;¿ES ESTO UN RETORNO DE CARRO?
000D C8 00220 RET Z ;SI ES SI, FIN DE LA ENTRADA
000E 77 00230 LD (HL),A ;GUARDAR CARACTER EN EL BUFFER
000F 78 00240 LD A,B ;NUMERO DE CARACTERES LEIDOS
0010 B9 00250 CP C ;¿SUPERIOR A NUMERO MAXIMO?
0011 300E 00260 JR NC,ZW ;SI ES SI, NO ACEPTARLO
0013 7E 00270 LD A,(HL) ;CARACTER LEIDO
0014 23 00280 INC HL ;INCREMENTAR PUNTERO
0016 04 00290 INC B ;UN CARACTER LEIDO DE MAS
0016 CD6900 00300 CALL AFI ;VISUALIZAR ESTE CARACTER
0019 1C 00310 INC E ;COLUMNA SIGUIENTE
001A CB6B 00320 BIT 5,E ;¿ESTAMOS EN EL EXTREMO DE LA LINEA?
001C 2803 00330 JR Z,ZW ;SI NO
001E 14 00340 INC D ;LINEA SIGUIENTE
001F 1E00 00350 LD E,0 ;COLOCARSE AL INICIO DE LA LINEA
0021 3AA800 00360 ZW LD A,(ATRB) ;ATRIBUTO GRAFICO
0024 F8 00370 PUSH AF ;GUARDAR EL ATRIBUTO
0025 CBFF 00380 SET 7,A ;PARPADEO
0027 32A800 00390 LD (ATRB),A ;NUEVO ATRIBUTO DE PARPADEO
002A 3E20 00400 LD A,' ' ;ESPACIO
002C CD6900 00410 CALL AFI ;VISUALIZAR EL CURSOR
002F F1 00420 POP AF ;RECUPERAR ATRIBUTO
0030 32A800 00430 LD (ATRB),A ;RESTAURAR ATRIBUTO
0033 18CD 00440 JR ZY ;LEER EL CARACTER SIGUIENTE
0035 78 00450 LD A,B ;NUMERO DE CARACTERES LEIDOS
0036 B7 00460 OR A ;¿NULO?
0037 28E8 00470 JR Z,ZW ;SI ES SI, NO TRATAR LA DELETE
0039 08 00480 DEC B ;UN CARACTER DE MENOS
003A 2B 00490 DEC HL ;DECREMENTAR PUNTERO
003B 3B20 00500 LD A,' ' ;ESPACIO
003D CD6900 00510 CALL AFI ;BORRAR EL CURSOR
0040 1D 00520 DEC E ;COLUMNA PRECEDENTE
0041 CB7B 00530 BIT 7,E ;¿SALTO DE LINEA?
0043 2BDC 00540 JR Z,ZW ;NO
0045 15 00550 DEC D ;LINEA PRECEDENTE
0046 1E1F 00560 LD E,31 ;COLOCARSE AL FINAL DE LA LINEA
0048 18D7 00570 JR ZW ;CONTINUACION

```

#### 4.5. Los mandos de juego

Las empuñaduras de juego de marca AGF, comercializadas por DIRECO, se colocan en paralelo con algunas teclas del teclado. El hecho de accionar uno de los elementos de la palanca equivale a la pulsación de la tecla correspondiente en el teclado. Así podremos leer el estado de los mandos de juego de la misma manera que lo hace-

mos para conocer el estado de las teclas correspondientes del teclado, es decir, mediante los ports de entrada/salida asociados, cuando se desee una lectura instantánea (en el caso de los juegos), o mediante el subprograma KEY. El cuadro siguiente resume esta correspondencia:

Elemento del mando de juego	Izquierda jugador 1	Abejo jugador 1	Arriba jugador 1	Derecha jugador 1	Tira jugador 1	Izquierda jugador 2	Abejo jugador 2	Arriba jugador 2	Derecha jugador 2	Tira jugador 2
Tecla del teclado	5	6	7	8	Ø	T	Y	U	I	P

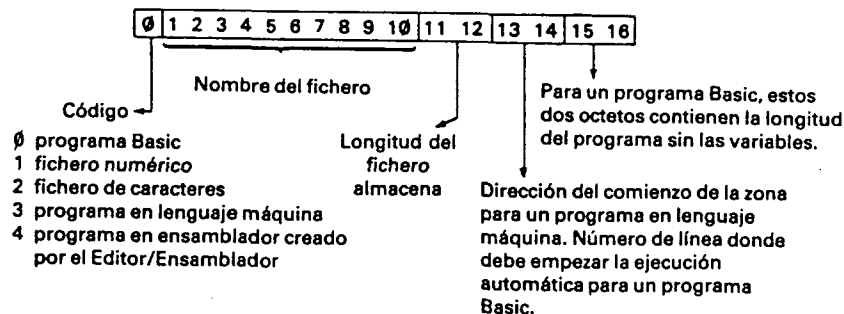
#### 4.6. La interfase de los cassettes

El bit 3 del port 254, utilizado como salida (OUT), acciona la salida de cassette, lo que tiene como efecto enviar un pulso por la salida MIC del SPECTRUM.

Inversamente, el bit 6 utilizado como entrada (IN) recibe una señal presente en la entrada EAR.

Estos dos bits se emplean por los subprogramas de escritura y de lectura en cassette que están situados en ROM. Estos dos subprogramas permiten leer o escribir en el cassette un número de octetos fijados por el usuario a partir de una posición fijada por él.

El subprograma de escritura se sitúa en la dirección 4C2H. Como parámetros IX deberá contener la dirección de inicio de la zona a transferir y DE el número de octetos a transferir. El registro A tendrá el valor Ø, o el valor FFH, según se desee escribir el encabezamiento del fichero o el cuerpo del mismo. Los ficheros están efectivamente divididos en dos partes, registradas mediante un formato algo diferente. La primera parte, o encabezamiento del fichero, contiene diversas informaciones relativas al fichero. Ocupa 17 octetos organizados de la forma siguiente:



La segunda parte contiene los datos propiamente dichos del fichero. Para un fichero que memoriza un programa en lenguaje máquina, esta parte contendrá todos los octetos del programa.

La serie de instrucciones que permiten la escritura en cassette de una de estas partes será la siguiente:

```
LD IX, dirección ; dirección de inicio de la zona que contiene
; los octetos a transferir.
LD DE, número ; número de octetos a transferir.
LD A, código ; ØØ o FFH, según la parte.
CALL 4C2H ; escritura.
```

El subprograma de lectura situado en la dirección 556H necesita los mismos valores como parámetros. Además, será necesario que el indicador Carry se coloque a 1. Las siguientes instrucciones permiten la lectura de una de las partes del fichero:

```
LD IX, dirección ; dirección de la zona donde serán
; transferidos los octetos leídos.
LD DE, número ; número de octetos a leer.
LD A, código ; ØØ o FFH, según la parte.
SCF ;
CALL 556H ; lectura.
```

## Anexo 1

### Las bases de numeración

Para contar nos hemos habituado a emplear la base 10 o base decimal. Esta base utiliza 10 símbolos representados por las cifras del 0 al 9. Para contar una sucesión de objetos enumeramos sucesivamente cada una de las cifras del 0 al 9. Cuando deseamos aumentar el número 9 en una unidad, colocamos el símbolo 1 delante de la cifra 9 que es reemplazada por 0, obtenemos así el número 10. La primera es la cifra de las decenas. Representa diez veces su valor habitual. La segunda cifra es añadida al resultado para obtener el número final. Así, un número decimal de dos cifras que se escriba  $xy$ , vale:  $10 \cdot x + y$ .

De la misma manera, un número decimal de tres cifras que se escriba  $xyz$ , vale  $100x + 10y + z$ .

Generalizando, un número decimal de  $n$  cifras que se escriba  $a_n a_{n-1} \dots a_1 a_0$  vale  $a_n \cdot 10^n + a_{n-1} \cdot 10^{n-1} + \dots + a_1 \cdot 10 + a^0$  donde  $10^k$ , es igual al producto de  $k$  veces el número 10.

$$10^k = \underbrace{10 \times 10 \times \dots \times 10}_{k \text{ veces}} \times 10$$

El número 10 de la base decimal representa el coeficiente multiplicativo que hay que utilizar para obtener el valor de una cifra, después del paso de una cifra a la posición adyacente a la izquierda en un número decimal. El número 10 es también igual al número de símbolos empleados en la base decimal.

La base 2, o base binaria, no utiliza más que dos símbolos representados por las cifras 0 y 1. El coeficiente multiplicativo evocado anteriormente vale 2.

En esta base los números de 0 a 10 en decimal se escriben respectivamente: 0, 1, 10, 11, 100, 101, 110, 111, 1000, 1001, 1010.

Un número binario que se escriba  $a_n a_{n-1} \dots a_1 a_0$  vale en decimal:

$$a_n \cdot 2^n + a_{n-1} \cdot 2^{n-1} + \dots + a_1 \cdot 2 + a_0$$

Así: 101 vale en decimal:  $1 \cdot 2^2 + 0 \cdot 2^1 + 1 = 4 + 1 = 5$

1010 vale en decimal:  $1 \cdot 2^3 + 0 \cdot 2^2 + 1 \cdot 2 + 0 = 8 + 2 = 10$

La base 2 que solamente posee dos símbolos se adapta bien a las tensiones con dos estados (0 V o 5 V) de la informática. No obstante, es poco manejable porque los números importantes necesitan una cantidad de símbolos demasiado grande para ser representados en esta base. Debido a ello se prefiere la base 16 o hexadecimal. Esta base utiliza 16 símbolos representados por las cifras de 0 a 9 más las letras de A a F. El coeficiente multiplicativo es igual a 16.

En esta base los números de 0 a 20 en decimal se escriben respectivamente: 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, F, 10, 11, 12, 13, 14.

Un número hexadecimal que se escriba  $a_n a_{n-1} \dots a_1 a_0$  vale en decimal:

$$a_n \cdot 16^n + a_{n-1} \cdot 16^{n-1} + \dots + a_1 \cdot 16 + a_0$$

Así: 3C vale en decimal:  $3 \cdot 16 + 12 = 60$

1E3A vale en decimal:  $1 \cdot 16^3 + 14 \cdot 16^2 + 3 \cdot 16 + 10 = 7738$

El interés de la base 16 proviene del hecho de que permite reagrupar cuatro símbolos binarios en un solo símbolo hexadecimal.

La siguiente tabla permite comprender mejor la correspondencia entre los números hexadecimales y los números binarios:

Decimal	Binario	Hexadecimal
0	0000	0
1	0001	1
2	0010	2
3	0011	3
4	0100	4
5	0101	5
6	0110	6
7	0111	7
8	1000	8
9	1001	9
10	1010	A
11	1011	B
12	1100	C
13	1101	D
14	1110	E
15	1111	F

Hay una correspondencia biunívoca entre un símbolo hexadecimal y un número binario cualquiera de 4 símbolos. De esto sacamos que la conversión de cualquier número binario en su representación

hexadecimal es muy rápida, agrupando los símbolos binarios de cuatro en cuatro.

Ejemplo:

1001	0010	1110	0111	binario
9	2	E	7	hexadecimal

La conversión de un número binario en su representación decimal es mucho menos rápida y necesita más cálculos.

Por lo tanto, en informática se utilizará sobre todo la base hexadecimal que permite conservar la estructura binaria de los números, siendo cuatro veces más concisos que con la base binaria.

## Anexo 2

### Lista de instrucciones del Z 80 clasificadas por códigos

NN designa un número de 16 bits; N un número de 8 bits; IND un índice, y DIS una distancia relativa.

Las instrucciones cuyo código empieza por CB, DD, ED, FD, estarán situadas al final del listado.

0000 00	00010	NOP	
0001 01EEFF	00020	LD	BC, NN
0004 02	00030	LD	(BC), A
0005 03	00040	INC	BC
0006 04	00050	INC	B
0007 05	00060	DEC	B
0008 060A	00070	LD	B, N
000A 07	00080	RLCA	
000B 08	00090	EX	AF, AF'
000C 09	00100	ADD	HL, BC
000D 0A	00110	LD	A, (BC)
000E 0B	00120	DEC	BC
000F 0C	00130	INC	C
0010 0D	00140	DEC	C
0011 0E0A	00150	LD	C, N
0013 0F	00160	RRCA	
0014 102E	00170	DJNZ	*, DIS
0016 11EEFF	00180	LD	DE, NN
0019 12	00190	LD	(DE), A
001A 13	00200	INC	DE
001B 14	00210	INC	D
001C 15	00220	DEC	D
001D 160A	00230	LD	D, N
001F 17	00240	RLA	
0020 182E	00250	JR	*, DIS
0022 19	00260	ADD	HL, DE
0023 1A	00270	LD	A, (DE)
0024 1B	00280	DEC	DE
0025 1C	00290	INC	E
0026 1D	00300	DEC	E
0027 1E0A	00310	LD	E, N
0029 1F	00320	RRA	
002A 202E	00330	JR	NZ, *, DIS
002C 21EEFF	00340	LD	HL, NN
002F 22EEFF	00350	LD	(NN), HL
0032 23	00360	INC	HL
0033 24	00370	INC	H
0034 25	00380	DEC	H
0035 260A	00390	LD	H, N
0037 27	00400	DAA	
0038 282E	00410	JR	Z, *, DIS
003A 29	00420	ADD	HL, HL
003B 2AEEFF	00430	LD	HL, (NN)
003E 2B	00440	DEC	HL
003F 2C	00450	INC	L
0040 2D	00460	DEC	L

0041	2EOA	00470	LD	L,N
0043	2F	00480	CPL	
0044	302E	00490	JR	NC, \$+DIS
0046	31EEFF	00500	LD	SP, NN
0049	32EEFF	00510	LD	(NN), A
004C	33	00520	INC	SP
004D	34	00530	INC	(HL)
004E	35	00540	DEC	(HL)
004F	360A	00550	LD	(HL), N
0051	37	00560	SCF	
0052	382E	00570	JR	C, \$+DIS
0054	39	00580	ADD	HL, SP
0055	3AEFF	00590	LD	A, (NN)
0058	3B	00600	DEC	SP
0059	3C	00610	INC	A
005A	3D	00620	DEC	A
005B	3EOA	00630	LD	A, N
005D	3F	00640	CCF	
005E	40	00650	LD	B, B
005F	41	00660	LD	B, C
0060	42	00670	LD	B, D
0061	43	00680	LD	B, E
0062	44	00690	LD	B, H
0063	45	00700	LD	B, L
0064	46	00710	LD	B, (HL)
0065	47	00720	LD	B, A
0066	48	00730	LD	C, B
0067	49	00740	LD	C, C
0068	4A	00750	LD	C, D
0069	4B	00760	LD	C, E
006A	4C	00770	LD	C, H
006B	4D	00780	LD	C, L
006C	4E	00790	LD	C, (HL)
006D	4F	00800	LD	C, A
006E	50	00810	LD	D, B
006F	51	00820	LD	D, C
0070	52	00830	LD	D, D
0071	53	00840	LD	D, E
0072	54	00850	LD	D, H
0073	55	00860	LD	D, L
0074	56	00870	LD	D, (HL)
0075	57	00880	LD	D, A
0076	58	00890	LD	E, B
0077	59	00900	LD	E, C
0078	5A	00910	LD	E, D
0079	5B	00920	LD	E, E
007A	5C	00930	LD	E, H
007B	5D	00940	LD	E, L
007C	5E	00950	LD	E, (HL)
007D	5F	00960	LD	E, A
007E	60	00970	LD	H, B
007F	61	00980	LD	H, C
0080	62	00990	LD	H, D
0081	63	01000	LD	H, E
0082	64	01010	LD	H, H
0083	65	01020	LD	H, L
0084	66	01030	LD	H, (HL)
0085	67	01040	LD	H, A
0086	68	01050	LD	L, B
0087	69	01060	LD	L, C
0088	6A	01070	LD	L, D
0089	6B	01080	LD	L, E
008A	6C	01090	LD	L, H
008B	6D	01100	LD	L, L
008C	6E	01110	LD	L, (HL)
008D	6F	01120	LD	L, A
008E	70	01130	LD	(HL), B
008F	71	01140	LD	(HL), C

0090	72	01150	LD	(HL), D
0091	73	01160	LD	(HL), E
0092	74	01170	LD	(HL), H
0093	75	01180	LD	(HL), L
0094	76	01190	HALT	
0095	77	01200	LD	(HL), A
0096	78	01210	LD	A, B
0097	79	01220	LD	A, C
0098	7A	01230	LD	A, D
0099	7B	01240	LD	A, E
009A	7C	01250	LD	A, H
009B	7D	01260	LD	A, L
009C	7E	01270	LD	A, (HL)
009D	7F	01280	LD	A, A
009E	80	01290	ADD	A, B
009F	81	01300	ADD	A, C
00A0	82	01310	ADD	A, D
00A1	83	01320	ADD	A, E
00A2	84	01330	ADD	A, H
00A3	85	01340	ADD	A, L
00A4	86	01350	ADD	A, (HL)
00A5	87	01360	ADD	A, A
00A6	88	01370	ADC	A, B
00A7	89	01380	ADC	A, C
00A8	8A	01390	ADC	A, D
00A9	8B	01400	ADC	A, E
00AA	8C	01410	ADC	A, H
00AB	8D	01420	ADC	A, L
00AC	8E	01430	ADC	A, (HL)
00AD	8F	01440	ADC	A, A
00AE	90	01450	SUB	B
00AF	91	01460	SUB	C
00B0	92	01470	SUB	D
00B1	93	01480	SUB	E
00B2	94	01490	SUB	H
00B3	95	01500	SUB	L
00B4	96	01510	SUB	(HL)
00B5	97	01520	SUB	A
00B6	98	01530	SBC	A, B
00B7	99	01540	SBC	A, C
00B8	9A	01550	SBC	A, D
00B9	9B	01560	SBC	A, E
00BA	9C	01570	SBC	A, H
00BB	9D	01580	SBC	A, L
00BC	9E	01590	SBC	A, (HL)
00BD	9F	01600	SBC	A, A
00BE	A0	01610	AND	B
00BF	A1	01620	AND	C
00C0	A2	01630	AND	D
00C1	A3	01640	AND	E
00C2	A4	01650	AND	H
00C3	A5	01660	AND	L
00C4	A6	01670	AND	(HL)
00C5	A7	01680	AND	A
00C6	A8	01690	XOR	B
00C7	A9	01700	XOR	C
00C8	AA	01710	XOR	D
00C9	AB	01720	XOR	E
00CA	AC	01730	XOR	H
00CB	AD	01740	XOR	L
00CC	AE	01750	XOR	(HL)
00CD	AF	01760	XOR	A
00CE	B0	01770	OR	B
00CF	B1	01780	OR	C
00D0	B2	01790	OR	D
00D1	B3	01800	OR	E
00D2	B4	01810	OR	H
00D3	B5	01820	OR	L

00D4 B6	01830	OR	(HL)
00D5 B7	01840	OR	A
00D6 B8	01850	CP	B
00D7 B9	01860	CP	C
00D8 BA	01870	CP	D
00D9 BB	01880	CP	E
00DA BC	01890	CP	H
00DB BD	01900	CP	L
00DC BE	01910	CP	(HL)
00DD BF	01920	CP	A
00DE C0	01930	RET	NZ
00DF C1	01940	POP	BC
00E0 C2EEFF	01950	JP	NZ, NN
00E3 C3EEFF	01960	JP	NN
00E6 C4EEFF	01970	CALL	NZ, NN
00E9 C5	01980	PUSH	BC
00EA C60A	01990	ADD	A, N
00EC C7	02000	RST	O
00ED C8	02010	RET	Z
00EE C9	02020	RET	
00EF CAEEFF	02030	JP	Z, NN
00F2 CCIEFF	02040	CALL	Z, NN
00F5 CDIEFF	02050	CALL	NN
00FB CE0A	02060	ADC	A, N
00FA CF	02070	RST	B
00FB D0	02080	RET	NC
00FC D1	02090	POP	DE
00FD D2EEFF	02100	JP	NC, NN
0100 D30A	02110	OUT	(N), A
0102 D4EEFF	02120	CALL	NC, NN
0105 D5	02130	PUSH	DE
0106 D60A	02140	SUB	N
0108 D7	02150	RST	10H
0109 D8	02160	RET	C
010A D9	02170	EXX	
010B DAEFF	02180	JP	C, NN
010E DBOA	02190	IN	A, (N)
0110 DCIEFF	02200	CALL	C, NN
0113 DE0A	02210	SBC	A, N
0115 DF	02220	RST	18H
0116 E0	02230	RET	PO
0117 E1	02240	POP	HL
0118 E2EEFF	02250	JP	PO, NN
011B E3	02260	EX	(SP), HL
011C E4EEFF	02270	CALL	PO, NN
011F E5	02280	PUSH	HL
0120 E60A	02290	AND	N
0122 E7	02300	RST	20H
0123 E8	02310	RET	PE
0124 E9	02320	JP	(HL)
0125 EAEEFF	02330	JP	PE, NN
0128 EB	02340	EX	DE, HL
0129 ECIEFF	02350	CALL	PE, NN
012C EE0A	02360	XOR	N
012E EF	02370	RST	28H
012F F0	02380	RET	P
0130 F1	02390	POP	AF
0131 F2EEFF	02400	JP	P, NN
0134 F3	02410	DI	
0135 F4EEFF	02420	CALL	P, NN
0138 F5	02430	PUSH	AF
0139 F60A	02440	OR	N
013B F7	02450	RST	30H
013C F8	02460	RET	M
013D F9	02470	LD	SP, HL
013E FAEEFF	02480	JP	M, NN
0141 FB	02490	EI	
0142 FCEEFF	02500	CALL	M, NN

0145 FE0A	02510	CP	N
0147 FF	02520	RST	38H
0148 CB00	02530	RLC	B
014A CB01	02540	RLC	C
014C CB02	02550	RLC	D
014E CB03	02560	RLC	E
0150 CB04	02570	RLC	H
0152 CB05	02580	RLC	L
0154 CB06	02590	RLC	(HL)
0156 CB07	02600	RLC	A
0158 CB08	02610	RRC	B
015A CB09	02620	RRC	C
015C CBOA	02630	RRC	D
015E CB0B	02640	RRC	E
0160 CBOC	02650	RRC	H
0162 CB0D	02660	RRC	L
0164 CBOE	02670	RRC	(HL)
0166 CBOF	02680	RRC	A
0168 CB10	02690	RL	B
016A CB11	02700	RL	C
016C CB12	02710	RL	D
016E CB13	02720	RL	E
0170 CB14	02730	RL	H
0172 CB15	02740	RL	L
0174 CB16	02750	RL	(HL)
0176 CB17	02760	RL	A
0178 CB18	02770	RR	B
017A CB19	02780	RR	C
017C CB1A	02790	RR	D
017E CB1B	02800	RR	E
0180 CB1C	02810	RR	H
0182 CB1D	02820	RR	L
0184 CB1E	02830	RR	(HL)
0186 CB1F	02840	RR	A
0188 CB20	02850	SLA	B
018A CB21	02860	SLA	C
018C CB22	02870	SLA	D
018E CB23	02880	SLA	E
0190 CB24	02890	SLA	H
0192 CB25	02900	SLA	L
0194 CB26	02910	SLA	(HL)
0196 CB27	02920	SLA	A
0198 CB28	02930	SRA	B
019A CB29	02940	SRA	C
019C CB2A	02950	SRA	D
019E CB2B	02960	SRA	E
01A0 CB2C	02970	SRA	H
01A2 CB2D	02980	SRA	L
01A4 CB2E	02990	SRA	(HL)
01A6 CB2F	03000	SRA	A
01A8 CB38	03010	SRL	B
01AA CB39	03020	SRL	C
01AC CB3A	03030	SRL	D
01AE CB3B	03040	SRL	E
01B0 CB3C	03050	SRL	H
01B2 CB3D	03060	SRL	L
01B4 CB3E	03070	SRL	(HL)
01B6 CB3F	03080	SRL	A
01B8 CB40	03090	BIT	O, B
01BA CB41	03100	BIT	O, C
01BC CB42	03110	BIT	O, D
01BE CB43	03120	BIT	O, E
01C0 CB44	03130	BIT	O, H
01C2 CB45	03140	BIT	O, L
01C4 CB46	03150	BIT	O, (HL)
01C6 CB47	03160	BIT	O, A
01C8 CB48	03170	BIT	1, B
01CA CB49	03180	BIT	1, C

01CC CB4A	03190	BIT	1, D
01CE CB4B	03200	BIT	1, E
01D0 CB4C	03210	BIT	1, H
01D2 CB4D	03220	BIT	1, L
01D4 CB4E	03230	BIT	1, (HL)
01D6 CB4F	03240	BIT	1, A
01D8 CB50	03250	BIT	2, B
01DA CB51	03260	BIT	2, C
01DC CB52	03270	BIT	2, D
01DE CB53	03280	BIT	2, E
01E0 CB54	03290	BIT	2, H
01E2 CB55	03300	BIT	2, L
01E4 CB56	03310	BIT	2, (HL)
01E6 CB57	03320	BIT	2, A
01E8 CB58	03330	BIT	3, B
01EA CB59	03340	BIT	3, C
01EC CB5A	03350	BIT	3, D
01EE CB5B	03360	BIT	3, E
01F0 CB5C	03370	BIT	3, H
01F2 CB5D	03380	BIT	3, L
01F4 CB5E	03390	BIT	3, (HL)
01F6 CB5F	03400	BIT	3, A
01F8 CB60	03410	BIT	4, B
01FA CB61	03420	BIT	4, C
01FC CB62	03430	BIT	4, D
01FE CB63	03440	BIT	4, E
0200 CB64	03450	BIT	4, H
0202 CB65	03460	BIT	4, L
0204 CB66	03470	BIT	4, (HL)
0206 CB67	03480	BIT	4, A
0208 CB68	03490	BIT	5, B
020A CB69	03500	BIT	5, C
020C CB6A	03510	BIT	5, D
020E CB6B	03520	BIT	5, E
0210 CB6C	03530	BIT	5, H
0212 CB6D	03540	BIT	5, L
0214 CB6E	03550	BIT	5, (HL)
0216 CB6F	03560	BIT	5, A
0218 CB70	03570	BIT	6, B
021A CB71	03580	BIT	6, C
021C CB72	03590	BIT	6, D
021E CB73	03600	BIT	6, E
0220 CB74	03610	BIT	6, H
0222 CB75	03620	BIT	6, L
0224 CB76	03630	BIT	6, (HL)
0226 CB77	03640	BIT	6, A
0228 CB78	03650	BIT	7, B
022A CB79	03660	BIT	7, C
022C CB7A	03670	BIT	7, D
022E CB7B	03680	BIT	7, E
0230 CB7C	03690	BIT	7, H
0232 CB7D	03700	BIT	7, L
0234 CB7E	03710	BIT	7, (HL)
0236 CB7F	03720	BIT	7, A
0238 CB80	03730	RES	0, B
023A CB81	03740	RES	0, C
023C CB82	03750	RES	0, D
023E CB83	03760	RES	0, E
0240 CB84	03770	RES	0, H
0242 CB85	03780	RES	0, L
0244 CB86	03790	RES	0, (HL)
0246 CB87	03800	RES	0, A
0248 CB88	03810	RES	1, B
024A CB89	03820	RES	1, C
024C CB8A	03830	RES	1, D
024E CB8B	03840	RES	1, E
0250 CB8C	03850	RES	1, H
0252 CB8D	03860	RES	1, L

0254 CB8E	03870	RES	1, (HL)
0256 CB8F	03880	RES	1, A
0258 CB90	03890	RES	2, B
025A CB91	03900	RES	2, C
025C CB92	03910	RES	2, D
025E CB93	03920	RES	2, E
0260 CB94	03930	RES	2, H
0262 CB95	03940	RES	2, L
0264 CB96	03950	RES	2, (HL)
0266 CB97	03960	RES	2, A
0268 CB98	03970	RES	3, B
026A CB99	03980	RES	3, C
026C CB9A	03990	RES	3, D
026E CB9B	04000	RES	3, E
0270 CB9C	04010	RES	3, H
0272 CB9D	04020	RES	3, L
0274 CB9E	04030	RES	3, (HL)
0276 CB9F	04040	RES	3, A
0278 CBA0	04050	RES	4, B
027A CBA1	04060	RES	4, C
027C CBA2	04070	RES	4, D
027E CBA3	04080	RES	4, E
0280 CBA4	04090	RES	4, H
0282 CBA5	04100	RES	4, L
0284 CBA6	04110	RES	4, (HL)
0286 CBA7	04120	RES	4, A
0288 CBA8	04130	RES	5, B
028A CBA9	04140	RES	5, C
028C CBAA	04150	RES	5, D
028E CBAB	04160	RES	5, E
0290 CBAC	04170	RES	5, H
0292 CBAD	04180	RES	5, L
0294 CBAE	04190	RES	5, (HL)
0296 CBAF	04200	RES	5, A
0298 CBB0	04210	RES	6, B
029A CBB1	04220	RES	6, C
029C CBB2	04230	RES	6, D
029E CBB3	04240	RES	6, E
02A0 CBB4	04250	RES	6, H
02A2 CBB5	04260	RES	6, L
02A4 CBB6	04270	RES	6, (HL)
02A6 CBB7	04280	RES	6, A
02A8 CBB8	04290	RES	7, B
02AA CBB9	04300	RES	7, C
02AC CBBA	04310	RES	7, D
02AE CBBB	04320	RES	7, E
02B0 CBBB	04330	RES	7, H
02B2 CBBD	04340	RES	7, L
02B4 CBBE	04350	RES	7, (HL)
02B6 CBBF	04360	RES	7, A
02B8 CBC0	04370	SET	0, B
02BA CBC1	04380	SET	0, C
02BC CBC2	04390	SET	0, D
02BE CBC3	04400	SET	0, E
02C0 CBC4	04410	SET	0, H
02C2 CBC5	04420	SET	0, L
02C4 CBC6	04430	SET	0, (HL)
02C6 CBC7	04440	SET	0, A
02C8 CBC8	04450	SET	1, B
02CA CBC9	04460	SET	1, C
02CC CBCA	04470	SET	1, D
02CE CBCB	04480	SET	1, E
02D0 CBCC	04490	SET	1, H
02D2 CBCE	04500	SET	1, L
02D4 CBCE	04510	SET	1, (HL)
02D6 CBCE	04520	SET	1, A
02D8 CBD0	04530	SET	2, B
02DA CBD1	04540	SET	2, C



02DC	CBD2	04550	SET	2, D
02DE	CBD3	04560	SET	2, E
02E0	CBD4	04570	SET	2, H
02E2	CBD5	04580	SET	2, L
02E4	CBD6	04590	SET	2, (HL)
02E6	CBD7	04600	SET	2, A
02E8	CBD8	04610	SET	3, B
02EA	CBD9	04620	SET	3, C
02EC	CBD4A	04630	SET	3, D
02EE	CBD8	04640	SET	3, E
02F0	CBD8C	04650	SET	3, H
02F2	CBD8D	04660	SET	3, L
02F4	CBDE	04670	SET	3, (HL)
02F6	CBDF	04680	SET	3, A
02F8	CBE0	04690	SET	4, B
02FA	CBE1	04700	SET	4, C
02FC	CBE2	04710	SET	4, D
02FE	CBE3	04720	SET	4, E
0300	CBE4	04730	SET	4, H
0302	CBE5	04740	SET	4, L
0304	CBE6	04750	SET	4, (HL)
0306	CBE7	04760	SET	4, A
0308	CBE8	04770	SET	5, B
030A	CBE9	04780	SET	5, C
030C	CBEA	04790	SET	5, D
030E	CBEB	04800	SET	5, E
0310	CBEC	04810	SET	5, H
0312	CBED	04820	SET	5, L
0314	CBEE	04830	SET	5, (HL)
0316	CBEF	04840	SET	5, A
0318	CBF0	04850	SET	6, B
031A	CBF1	04860	SET	6, C
031C	CBF2	04870	SET	6, D
031E	CBF3	04880	SET	6, E
0320	CBF4	04890	SET	6, H
0322	CBF5	04900	SET	6, L
0324	CBF6	04910	SET	6, (HL)
0326	CBF7	04920	SET	6, A
0328	CBF8	04930	SET	7, B
032A	CBF9	04940	SET	7, C
032C	CBFA	04950	SET	7, D
032E	CBFB	04960	SET	7, E
0330	CBFC	04970	SET	7, H
0332	CBFD	04980	SET	7, L
0334	CBFE	04990	SET	7, (HL)
0336	CBFF	05000	SET	7, A
0338	DD09	05010	ADD	IX, BC
033A	DD19	05020	ADD	IX, DE
033C	DD21EEFF	05030	LD	IX, NN
0340	DD22EEFF	05040	LD	(NN), IX
0344	DD23	05050	INC	IX
0346	DD29	05060	ADD	IX, IX
0348	DD2AEEFF	05070	LD	IX, (NN)
034C	DD2B	05080	DEC	IX
034E	DD3405	05090	INC	(IX+IND)
0351	DD3505	05100	DEC	(IX+IND)
0354	DD36050A	05110	LD	(IX+IND), N
0358	DD39	05120	ADD	IX, SP
035A	DD4605	05130	LD	B, (IX+IND)
035D	DD4E05	05140	LD	C, (IX+IND)
0360	DD5605	05150	LD	D, (IX+IND)
0363	DD5E05	05160	LD	E, (IX+IND)
0366	DD6605	05170	LD	H, (IX+IND)
0369	DD6E05	05180	LD	L, (IX+IND)
036C	DD7005	05190	LD	(IX+IND), B
036F	DD7105	05200	LD	(IX+IND), C
0372	DD7205	05210	LD	(IX+IND), D
0375	DD7305	05220	LD	(IX+IND), E

0378	DD7405	05230	LD	(IX+IND), H
037B	DD7505	05240	LD	(IX+IND), L
037E	DD7705	05250	LD	(IX+IND), A
0381	DD7E05	05260	LD	A, (IX+IND)
0384	DD8605	05270	ADD	A, (IX+IND)
0387	DD8E05	05280	ADC	A, (IX+IND)
038A	DD9605	05290	SUB	(IX+IND)
038D	DD9E05	05300	SBC	A, (IX+IND)
0390	DDA605	05310	AND	(IX+IND)
0393	DDAE05	05320	XOR	(IX+IND)
0396	DDB605	05330	OR	(IX+IND)
0399	DDBE05	05340	CP	(IX+IND)
039C	DDE1	05350	POP	IX
039E	DDE3	05360	EX	(SP), IX
03A0	DDE5	05370	IX	PUSH
03A2	DDE9	05380	JP	(IX)
03A4	DDF9	05390	LD	SP, IX
03A6	DDCB0506	05400	RLC	(IX+IND)
03AA	DDCB050E	05410	RRC	(IX+IND)
03AE	DDCB0516	05420	RL	(IX+IND)
03B2	DDCB051E	05430	RR	(IX+IND)
03B6	DDCB0526	05440	SLA	(IX+IND)
03BA	DDCB052E	05450	SRA	(IX+IND)
03BE	DDCB053E	05460	SRL	(IX+IND)
03C2	DDCB0546	05470	BIT	0, (IX+IND)
03C6	DDCB054E	05480	BIT	1, (IX+IND)
03CA	DDCB0556	05490	BIT	2, (IX+IND)
03CE	DDCB055E	05500	BIT	3, (IX+IND)
03D2	DDCB0566	05510	BIT	4, (IX+IND)
03D6	DDCB056E	05520	BIT	5, (IX+IND)
03DA	DDCB0576	05530	BIT	6, (IX+IND)
03DE	DDCB057E	05540	BIT	7, (IX+IND)
03E2	DDCB0586	05550	RES	0, (IX+IND)
03E6	DDCB058E	05560	RES	1, (IX+IND)
03EA	DDCB0596	05570	RES	2, (IX+IND)
03EE	DDCB059E	05580	RES	3, (IX+IND)
03F2	DDCB05A6	05590	RES	4, (IX+IND)
03F6	DDCB05AE	05600	RES	5, (IX+IND)
03FA	DDCB05B6	05610	RES	6, (IX+IND)
03FE	DDCB05BE	05620	RES	7, (IX+IND)
0402	DDCB05C6	05630	SET	0, (IX+IND)
0406	DDCB05CE	05640	SET	1, (IX+IND)
040A	DDCB05D6	05650	SET	2, (IX+IND)
040E	DDCB05DE	05660	SET	3, (IX+IND)
0412	DDCB05E6	05670	SET	4, (IX+IND)
0416	DDCB05EE	05680	SET	5, (IX+IND)
041A	DDCB05F6	05690	SET	6, (IX+IND)
041E	DDCB05FE	05700	SET	7, (IX+IND)
0422	ED40	05710	IN	B, (C)
0424	ED41	05720	OUT	(C), B
0426	ED42	05730	SBC	HL, BC
0428	ED43EEFF	05740	LD	(NN), BC
042C	ED44	05750	NEG	
042E	ED45	05760	RETN	
0430	ED46	05770	IM	0
0432	ED47	05780	LD	I, A
0434	ED48	05790	IN	C, (C)
0436	ED49	05800	OUT	(C), C
0438	ED4A	05810	ADC	HL, BC
043A	ED4BEEFF	05820	LD	BC, (NN)
043E	ED4D	05830	RETI	
0440	ED4F	05840	LD	R, A
0442	ED50	05850	IN	D, (C)
0444	ED51	05860	OUT	(C), D
0446	ED52	05870	SBC	HL, DE
0448	ED53EEFF	05880	LD	(NN), DE
044C	ED56	05890	IM	1
044E	ED57	05900	LD	A, I

0450	ED58	05910	IN	E, (C)
0452	ED59	05920	OUT	(C), E
0454	ED5A	05930	ADC	HL, DE
0456	ED5BEEFF	05940	LD	DE, (NN)
045A	ED5E	05950	IM	Z
045C	ED5F	05960	LD	A, R
045E	ED60	05970	IN	H, (C)
0460	ED61	05980	OUT	(C), H
0462	ED62	05990	SBC	HL, HL
0464	ED67	06000	RRD	
0466	ED68	06010	IN	L, (C)
0468	ED69	06020	OUT	(C), L
046A	ED6A	06030	ADC	HL, HL
046C	ED6F	06040	RLD	
046E	ED72	06050	SBC	HL, SP
0470	ED73EEFF	06060	LD	(NN), SP
0474	ED7B	06070	IN	A, (C)
0476	ED79	06080	OUT	(C), A
0478	ED7A	06090	ADC	HL, SP
047A	ED7BEEFF	06100	LD	SP, (NN)
047E	EDA0	06110	LDI	
0480	EDA1	06120	CPI	
0482	EDA2	06130	INI	
0484	EDA3	06140	OUTI	
0486	EDA8	06150	LDD	
0488	EDA9	06160	CPD	
048A	EDAA	06170	IND	
048C	EDAB	06180	OUTD	
048E	EDB0	06190	LDIR	
0490	EDB1	06200	CPIR	
0492	EDB2	06210	INIR	
0494	EDB3	06220	OTIR	
0496	EDB8	06230	LDDR	
0498	EDB9	06240	CPDR	
049A	EDBA	06250	INDR	
049C	EDBB	06260	OTDR	
049E	FD09	06270	ADD	IY, BC
04A0	FD19	06280	ADD	IY, DE
04A2	FD21EEFF	06290	LD	IY, NN
04A6	FD22EEFF	06300	LD	(NN), IY
04AA	FD23	06310	INC	IY
04AC	FD29	06320	ADD	IY, IY
04AE	FD2AEEFF	06330	LD	IY, (NN)
04B2	FD2B	06340	DEC	IY
04B4	FD3405	06350	INC	(IY+IND)
04B7	FD3505	06360	DEC	(IY+IND)
04BA	FD36050A	06370	LD	(IY+IND), N
04BE	FD39	06380	ADD	IY, SP
04C0	FD4605	06390	LD	B, (IY+IND)
04C3	FD4E05	06400	LD	C, (IY+IND)
04C6	FD5605	06410	LD	D, (IY+IND)
04C9	FD5E05	06420	LD	E, (IY+IND)
04CC	FD6605	06430	LD	H, (IY+IND)
04CF	FD6E05	06440	LD	L, (IY+IND)
04D2	FD7005	06450	LD	(IY+IND), B
04D5	FD7105	06460	LD	(IY+IND), C
04D8	FD7205	06470	LD	(IY+IND), D
04DB	FD7305	06480	LD	(IY+IND), E
04DE	FD7405	06490	LD	(IY+IND), H
04E1	FD7505	06500	LD	(IY+IND), L
04E4	FD7705	06510	LD	(IY+IND), A
04E7	FD7E05	06520	LD	A, (IY+IND)
04EA	FD8605	06530	ADD	A, (IY+IND)
04ED	FD8E05	06540	ADC	A, (IY+IND)
04F0	FD9605	06550	SUB	(IY+IND)
04F3	FD9E05	06560	SBC	A, (IY+IND)
04F6	FDA605	06570	AND	(IY+IND)
04F9	FDAE05	06580	XOR	(IY+IND)

04FC	FDB605	06590	OR	(IY+IND)
04FF	FDBE05	06600	CP	(IY+IND)
0502	FDE1	06610	POP	IY
0504	FDE3	06620	EX	(SP), IY
0506	FDE5	06630	PUSH	IY
0508	FDE9	06640	JP	(IY)
050A	FD99	06650	LD	SP, IY
050C	FDCB0506	06660	RLC	(IY+IND)
0510	FDCB050E	06670	RRC	(IY+IND)
0514	FDCB0516	06680	RL	(IY+IND)
0518	FDCB051E	06690	RR	(IY+IND)
051C	FDCB0526	06700	SLA	(IY+IND)
0520	FDCB052E	06710	SRA	(IY+IND)
0524	FDCB053E	06720	SRL	(IY+IND)
0528	FDCB0546	06730	BIT	0, (IY+IND)
052C	FDCB054E	06740	BIT	1, (IY+IND)
0530	FDCB0556	06750	BIT	2, (IY+IND)
0534	FDCB055E	06760	BIT	3, (IY+IND)
0538	FDCB0566	06770	BIT	4, (IY+IND)
053C	FDCB056E	06780	BIT	5, (IY+IND)
0540	FDCB0576	06790	BIT	6, (IY+IND)
0544	FDCB057E	06800	BIT	7, (IY+IND)
0548	FDCB0586	06810	RES	0, (IY+IND)
054C	FDCB058E	06820	RES	1, (IY+IND)
0550	FDCB0596	06830	RES	2, (IY+IND)
0554	FDCB059E	06840	RES	3, (IY+IND)
0558	FDCB05A6	06850	RES	4, (IY+IND)
055C	FDCB05AE	06860	RES	5, (IY+IND)
0560	FDCB05B6	06870	RES	6, (IY+IND)
0564	FDCB05BE	06880	RES	7, (IY+IND)
0568	FDCB05C6	06890	SET	0, (IY+IND)
056C	FDCB05CE	06900	SET	1, (IY+IND)
0570	FDCB05D6	06910	SET	2, (IY+IND)
0574	FDCB05DE	06920	SET	3, (IY+IND)
0578	FDCB05E6	06930	SET	4, (IY+IND)
057C	FDCB05EE	06940	SET	5, (IY+IND)
0580	FDCB05F6	06950	SET	6, (IY+IND)
0584	FDCB05FE	06960	SET	7, (IY+IND)
FFEE	06970	NN	EQU	OFFEEH
0005	06980	IND	EQU	5
000A	06990	N	EQU	10
0030	07000	DIS	EQU	30H
0000	07010		END	

# Anexo 3

## Lista de instrucciones del Z 80

### clasificadas por mnemotécnicos

NN designa un número de 16 bits; N un número de 8 bits; IND un índice y DIS una distancia relativa.

Dirección	Código	Línea	Mnemotécnico
0000	BE	00010	ADC A, (HL)
0001	DD8E05	00020	ADC A, (IX+IND)
0004	FD8E05	00030	ADC A, (IY+IND)
0007	BF	00040	ADC A, A
0008	88	00050	ADC A, B
0009	89	00060	ADC A, C
000A	8A	00070	ADC A, D
000B	8B	00080	ADC A, E
000C	8C	00090	ADC A, H
000D	8D	00100	ADC A, L
000E	CE10	00110	ADC A, N
0010	ED4A	00120	ADC HL, BC
0012	ED5A	00130	ADC HL, DE
0014	ED6A	00140	ADC HL, HL
0016	ED7A	00150	ADC HL, SP
0018	86	00160	ADD A, (HL)
0019	DD8605	00170	ADD A, (IX+IND)
001C	FD8605	00180	ADD A, (IY+IND)
001F	87	00190	ADD A, A
0020	80	00200	ADD A, B
0021	81	00210	ADD A, C
0022	82	00220	ADD A, D
0023	83	00230	ADD A, E
0024	84	00240	ADD A, H
0025	85	00250	ADD A, L
0026	C610	00260	ADD A, N
0028	09	00270	ADD HL, BC
0029	19	00280	ADD HL, DE
002A	29	00290	ADD HL, HL
002B	39	00300	ADD HL, SP
002C	DD09	00310	ADD IX, BC
002E	DD19	00320	ADD IX, DE
0030	DD29	00330	ADD IX, IX
0032	DD39	00340	ADD IX, SP
0034	FD09	00350	ADD IY, BC
0036	FD19	00360	ADD IY, DE
0038	FD29	00370	ADD IY, IY
003A	FD39	00380	ADD IY, SP
003C	A6	00390	AND (HL)
003D	DDA605	00400	AND (IX+IND)
0040	FDA605	00410	AND (IY+IND)
0043	A7	00420	AND A
0044	A0	00430	AND B
0045	A1	00440	AND C
0046	A2	00450	AND D

0047	A3	00460	AND E
0048	A4	00470	AND H
0049	A5	00480	AND L
004A	E610	00490	AND N
004C	CB46	00500	BIT 0, (HL)
004E	DDCB0546	00510	BIT 0, (IX+IND)
0052	FDCB0546	00520	BIT 0, (IY+IND)
0056	CB47	00530	BIT 0, A
0058	CB40	00540	BIT 0, B
005A	CB41	00550	BIT 0, C
005C	CB42	00560	BIT 0, D
005E	CB43	00570	BIT 0, E
0060	CB44	00580	BIT 0, H
0062	CB45	00590	BIT 0, L
0064	CB4E	00600	BIT 1, (HL)
0066	DDCB054E	00610	BIT 1, (IX+IND)
006A	FDCB054E	00620	BIT 1, (IY+IND)
006E	CB4F	00630	BIT 1, A
0070	CB48	00640	BIT 1, B
0072	CB49	00650	BIT 1, C
0074	CB4A	00660	BIT 1, D
0076	CB4B	00670	BIT 1, E
007B	CB4C	00680	BIT 1, H
007A	CB4D	00690	BIT 1, L
007C	CB56	00700	BIT 2, (HL)
007E	DDCB0556	00710	BIT 2, (IX+IND)
0082	FDCB0556	00720	BIT 2, (IY+IND)
0086	CB57	00730	BIT 2, A
0088	CB50	00740	BIT 2, B
008A	CB51	00750	BIT 2, C
008C	CB52	00760	BIT 2, D
008E	CB53	00770	BIT 2, E
0090	CB54	00780	BIT 2, H
0092	CB55	00790	BIT 2, L
0094	CB5E	00800	BIT 3, (HL)
0096	DDCB055E	00810	BIT 3, (IX+IND)
009A	FDCB055E	00820	BIT 3, (IY+IND)
009E	CB5F	00830	BIT 3, A
00A0	CB58	00840	BIT 3, B
00A2	CB59	00850	BIT 3, C
00A4	CB5A	00860	BIT 3, D
00A6	CB5B	00870	BIT 3, E
00A8	CB5C	00880	BIT 3, H
00AA	CB5D	00890	BIT 3, L
00AC	CB66	00900	BIT 4, (HL)
00AE	DDCB0566	00910	BIT 4, (IX+IND)
00B2	FDCB0566	00920	BIT 4, (IY+IND)
00B6	CB67	00930	BIT 4, A
00B8	CB60	00940	BIT 4, B
00BA	CB61	00950	BIT 4, C
00BC	CB62	00960	BIT 4, D
00BE	CB63	00970	BIT 4, E
00C0	CB64	00980	BIT 4, H
00C2	CB65	00990	BIT 4, L
00C4	CB6E	01000	BIT 5, (HL)
00C6	DDCB056E	01010	BIT 5, (IX+IND)
00CA	FDCB056E	01020	BIT 5, (IY+IND)
00CE	CB6F	01030	BIT 5, A
00D0	CB68	01040	BIT 5, B
00D2	CB69	01050	BIT 5, C
00D4	CB6A	01060	BIT 5, D
00D6	CB6B	01070	BIT 5, E
00D8	CB6C	01080	BIT 5, H
00DA	CB6D	01090	BIT 5, L
00DC	CB76	01100	BIT 6, (HL)
00DE	DDCB0576	01110	BIT 6, (IX+IND)
00E2	FDCB0576	01120	BIT 6, (IY+IND)
00E6	CB77	01130	BIT 6, A

00EB CB70	01140	BIT	6,B
00EA CB71	01150	BIT	6,C
00EC CB72	01160	BIT	6,D
00EE CB73	01170	BIT	6,E
00F0 CB74	01180	BIT	6,H
00F2 CB75	01190	BIT	6,L
00F4 CB7E	01200	BIT	7,(HL)
00F6 DDCB057E	01210	BIT	7,(IX+IND)
00FA FDCB057E	01220	BIT	7,(IY+IND)
00FE CB7F	01230	BIT	7,A
0100 CB78	01240	BIT	7,B
0102 CB79	01250	BIT	7,C
0104 CB7A	01260	BIT	7,D
0106 CB7B	01270	BIT	7,E
0108 CB7C	01280	BIT	7,H
010A CB7D	01290	BIT	7,L
010C DCEEFF	01300	CALL	C,NN
010F FCEEFF	01310	CALL	M,NN
0112 D4EEFF	01320	CALL	NC,NN
0115 CDEEFF	01330	CALL	NN
0118 C4EEFF	01340	CALL	NZ,NN
011B F4EEFF	01350	CALL	P,NN
011E ECEEFF	01360	CALL	PE,NN
0121 E4EEFF	01370	CALL	PO,NN
0124 CCEEFF	01380	CALL	Z,NN
0127 3F	01390	CCF	
012B BE	01400	CP	(HL)
0129 DDBE05	01410	CP	(IX+IND)
012C FDBE05	01420	CP	(IY+IND)
012F BF	01430	CP	A
0130 B8	01440	CP	B
0131 B9	01450	CP	C
0132 BA	01460	CP	D
0133 BB	01470	CP	E
0134 BC	01480	CP	H
0135 BD	01490	CP	L
0136 FE10	01500	CP	N
0138 EDA9	01510	CPD	
013A EDB9	01520	CPDR	
013C EDA1	01530	CPI	
013E EDB1	01540	CPIR	
0140 2F	01550	CPL	
0141 27	01560	DAA	
0142 35	01570	DEC	(HL)
0143 DD3505	01580	DEC	(IX+IND)
0146 FD3505	01590	DEC	(IY+IND)
0149 3D	01600	DEC	A
014A 05	01610	DEC	B
014B 0B	01620	DEC	BC
014C 0D	01630	DEC	C
014D 15	01640	DEC	D
014E 1B	01650	DEC	DE
014F 1D	01660	DEC	E
0150 25	01670	DEC	H
0151 2B	01680	DEC	HL
0152 DD2B	01690	DEC	IX
0154 FD2B	01700	DEC	IY
0156 2D	01710	DEC	L
0157 3B	01720	DEC	SP
0158 F3	01730	DI	
0159 102E	01740	DJNZ	*+DIS
015B FB	01750	EI	
015C E3	01760	EX	(SP),HL
015D DDEZ	01770	EX	(SP),IX
015F FDE3	01780	EX	(SP),IY
0161 0B	01790	EX	AF,AF'
0162 EB	01800	EX	DE,HL
0163 D9	01810	EXX	

0164 7E	01820	HALT	
0165 ED46	01830	IM	0
0167 ED56	01840	IM	1
0169 ED5E	01850	IM	2
016B ED78	01860	IN	A,(C)
016D DB10	01870	IN	A,(N)
016F ED40	01880	IN	B,(C)
0171 ED48	01890	IN	C,(C)
0173 ED50	01900	IN	D,(C)
0175 ED58	01910	IN	E,(C)
0177 ED60	01920	IN	H,(C)
0179 ED68	01930	IN	L,(C)
017B 34	01940	INC	(HL)
017C DD3405	01950	INC	(IX+IND)
017F FD3405	01960	INC	(IY+IND)
0182 3C	01970	INC	A
0183 04	01980	INC	B
0184 03	01990	INC	BC
0185 0C	02000	INC	C
0186 14	02010	INC	D
0187 13	02020	INC	DE
0188 1C	02030	INC	E
0189 24	02040	INC	H
018A 23	02050	INC	HL
018B DD23	02060	INC	IX
018D FD23	02070	INC	IY
018F 2C	02080	INC	L
0190 33	02090	INC	SP
0191 EDAA	02100	IND	
0193 ED8A	02110	INDR	
0195 EDA2	02120	INI	
0197 EDB2	02130	INIR	
0199 E9	02140	JP	(HL)
019A DDE9	02150	JP	(IX)
019C FDE9	02160	JP	(IY)
019E DAEFF	02170	JF	C,NN
01A1 FAEFF	02180	JP	M,NN
01A4 D2EEFF	02190	JP	NC,NN
01A7 C3EEFF	02200	JP	NN
01AA C2EEFF	02210	JP	NZ,NN
01AD F2EEFF	02220	JP	P,NN
01B0 EAEFF	02230	JP	PE,NN
01B3 E2EEFF	02240	JP	PO,NN
01B6 CAEEFF	02250	JP	Z,NN
01B9 3B2E	02260	JR	C,*+DIS
01BB 1B2E	02270	JR	*+DIS
01BD 302E	02280	JR	NC,*+DIS
01BF 202E	02290	JR	NZ,*+DIS
01C1 2B2E	02300	JR	Z,*+DIS
01C3 02	02310	LD	(BC),A
01C4 12	02320	LD	(DE),A
01C5 77	02330	LD	(HL),A
01C6 70	02340	LD	(HL),B
01C7 71	02350	LD	(HL),C
01C8 72	02360	LD	(HL),D
01C9 73	02370	LD	(HL),E
01CA 74	02380	LD	(HL),H
01CB 75	02390	LD	(HL),L
01CC 3610	02400	LD	(HL),N
01CE DD7705	02410	LD	(IX+IND),A
01D1 DD7005	02420	LD	(IX+IND),B
01D4 DD7105	02430	LD	(IX+IND),C
01D7 DD7205	02440	LD	(IX+IND),D
01DA DD7305	02450	LD	(IX+IND),E
01DD DD7405	02460	LD	(IX+IND),H
01E0 DD7505	02470	LD	(IX+IND),L
01E3 DD360510	02480	LD	(IX+IND),N
01E7 FD7705	02490	LD	(IY+IND),A

01EA	FD7005	02500	LD	(IY+IND), B
01ED	FD7105	02510	LD	(IY+IND), C
01FO	FD7205	02520	LD	(IY+IND), D
01F3	FD7305	02530	LD	(IY+IND), E
01F6	FD7405	02540	LD	(IY+IND), H
01F9	FD7505	02550	LD	(IY+IND), L
01FC	FD360510	02560	LD	(IY+IND), N
0200	32EEFF	02570	LD	(NN), A
0203	ED43EEFF	02580	LD	(NN), BC
0207	ED53EEFF	02590	LD	(NN), DE
020B	22EEFF	02600	LD	(NN), HL
020E	DD22EEFF	02610	LD	(NN), IX
0212	FD22EEFF	02620	LD	(NN), IY
0216	ED73EEFF	02630	LD	(NN), SP
021A	0A	02640	LD	A, (BC)
021B	1A	02650	LD	A, (DE)
021C	7E	02660	LD	A, (HL)
021D	DD7E05	02670	LD	A, (IX+IND)
0220	FD7E05	02680	LD	A, (IY+IND)
0223	3AEEFF	02690	LD	A, (NN)
0226	7F	02700	LD	A, A
0227	7B	02710	LD	A, B
0228	79	02720	LD	A, C
0229	7A	02730	LD	A, D
022A	7B	02740	LD	A, E
022B	7C	02750	LD	A, H
022C	ED57	02760	LD	A, I
022E	7D	02770	LD	A, L
022F	3E10	02780	LD	A, N
0231	ED5F	02790	LD	A, R
0233	46	02800	LD	B, (HL)
0234	DD4605	02810	LD	B, (IX+IND)
0237	FD4605	02820	LD	B, (IY+IND)
023A	47	02830	LD	B, A
023B	40	02840	LD	B, B
023C	41	02850	LD	B, C
023D	42	02860	LD	B, D
023E	43	02870	LD	B, E
023F	44	02880	LD	B, H
0240	45	02890	LD	B, L
0241	0610	02900	LD	B, N
0243	ED4BEEFF	02910	LD	BC, (NN)
0247	01EEFF	02920	LD	BC, NN
024A	4E	02930	LD	C, (HL)
024B	DD4E05	02940	LD	C, (IX+IND)
024E	FD4E05	02950	LD	C, (IY+IND)
0251	4F	02960	LD	C, A
0252	48	02970	LD	C, B
0253	49	02980	LD	C, C
0254	4A	02990	LD	C, D
0255	4B	03000	LD	C, E
0256	4C	03010	LD	C, H
0257	4D	03020	LD	C, L
0258	0E10	03030	LD	C, N
025A	56	03040	LD	D, (HL)
025B	DD5605	03050	LD	D, (IX+IND)
025E	FD5605	03060	LD	D, (IY+IND)
0261	57	03070	LD	D, A
0262	50	03080	LD	D, B
0263	51	03090	LD	D, C
0264	52	03100	LD	D, D
0265	53	03110	LD	D, E
0266	54	03120	LD	D, H
0267	55	03130	LD	D, L
0268	1610	03140	LD	D, N
026A	ED5BEEFF	03150	LD	DE, (NN)
026E	11EEFF	03160	LD	DE, NN
0271	5E	03170	LD	E, (HL)

0272	DD5E05	03180	LD	E, (IX+IND)
0275	FD5E05	03190	LD	E, (IY+IND)
0278	5F	03200	LD	E, A
0279	58	03210	LD	E, B
027A	59	03220	LD	E, C
027B	5A	03230	LD	E, D
027C	5B	03240	LD	E, E
027D	5C	03250	LD	E, H
027E	5D	03260	LD	E, L
027F	1E10	03270	LD	E, N
0281	66	03280	LD	H, (HL)
0282	DD6605	03290	LD	H, (IX+IND)
0285	FD6605	03300	LD	H, (IY+IND)
0288	67	03310	LD	H, A
0289	60	03320	LD	H, B
028A	61	03330	LD	H, C
028B	62	03340	LD	H, D
028C	63	03350	LD	H, E
028D	64	03360	LD	H, H
028E	65	03370	LD	H, L
028F	2610	03380	LD	H, N
0291	2AEEFF	03390	LD	HL, (NN)
0294	21EEFF	03400	LD	HL, NN
0297	ED47	03410	LD	I, A
0299	DD2AEEFF	03420	LD	IX, (NN)
029D	DD21EEFF	03430	LD	IX, NN
02A1	FD2AEEFF	03440	LD	IY, (NN)
02A5	FD21EEFF	03450	LD	IY, NN
02A9	6E	03460	LD	L, (HL)
02AA	DD6E05	03470	LD	L, (IX+IND)
02AD	FD6E05	03480	LD	L, (IY+IND)
02B0	6F	03490	LD	L, A
02B1	68	03500	LD	L, B
02B2	69	03510	LD	L, C
02B3	6A	03520	LD	L, D
02B4	6B	03530	LD	L, E
02B5	6C	03540	LD	L, H
02B6	6D	03550	LD	L, L
02B7	2E10	03560	LD	L, N
02B9	ED4F	03570	LD	R, A
02BB	ED7BEEFF	03580	LD	SP, (NN)
02BF	F9	03590	LD	SP, HL
02C0	DDF9	03600	LD	SP, IX
02C2	FDF9	03610	LD	SP, IY
02C4	31EEFF	03620	LD	SP, NN
02C7	EDAB	03630	LDD	
02C9	EDB8	03640	LDDR	
02CB	EDA0	03650	LDI	
02CD	EDB0	03660	LDIF	
02CF	ED44	03670	NEG	
02D1	00	03680	NOP	
02D2	B6	03690	OR	(HL)
02D3	DD6605	03700	OR	(IX+IND)
02D6	FDB605	03710	OR	(IY+IND)
02D9	B7	03720	OR	A
02DA	B0	03730	OR	B
02DB	B1	03740	OR	C
02DC	B2	03750	OR	D
02DD	B3	03760	OR	E
02DE	B4	03770	OR	H
02DF	B5	03780	OR	L
02E0	F610	03790	OR	N
02E2	EDB8	03800	OTDR	
02E4	EDB3	03810	OTIR	
02E6	ED79	03820	OUT	(C), A
02E8	ED41	03830	OUT	(C), B
02EA	ED49	03840	OUT	(C), C
02EC	ED51	03850	OUT	(C), D

02EE	ED59	03860	OUT	(C),E
02F0	ED61	03870	OUT	(C),H
02F2	ED69	03880	OUT	(C),L
02F4	D310	03890	OUT	(N),A
02F6	EDAB	03900	OUTD	
02F8	EDA3	03910	OUTI	
02FA	F1	03920	POP	AF
02FB	C1	03930	POP	BC
02FC	D1	03940	POP	DE
02FD	E1	03950	POP	HL
02FE	DDE1	03960	POP	IX
0300	FDE1	03970	POP	IY
0302	F5	03980	PUSH	AF
0303	C5	03990	PUSH	BC
0304	D5	04000	PUSH	DE
0305	E5	04010	PUSH	HL
0306	DDE5	04020	PUSH	IX
0308	FDE5	04030	PUSH	IY
030A	C8B6	04040	RES	0, (HL)
030C	DDCB05B6	04050	RES	0, (IX+IND)
0310	FDCB05B6	04060	RES	0, (IY+IND)
0314	CB87	04070	RES	0,A
0316	CB80	04080	RES	0,B
0318	CB81	04090	RES	0,C
031A	CB82	04100	RES	0,D
031C	CB83	04110	RES	0,E
031E	CB84	04120	RES	0,H
0320	CB85	04130	RES	0,L
0322	CB8E	04140	RES	1, (HL)
0324	DDCB05BE	04150	RES	1, (IX+IND)
0328	FDCB05BE	04160	RES	1, (IY+IND)
032C	CB8F	04170	RES	1,A
032E	CB88	04180	RES	1,B
0330	CB89	04190	RES	1,C
0332	CB8A	04200	RES	1,D
0334	CB8B	04210	RES	1,E
0336	CB8C	04220	RES	1,H
0338	CB8D	04230	RES	1,L
033A	CB96	04240	RES	2, (HL)
033C	DDCB0596	04250	RES	2, (IX+IND)
0340	FDCB0596	04260	RES	2, (IY+IND)
0344	CB97	04270	RES	2,A
0346	CB90	04280	RES	2,B
0348	CB91	04290	RES	2,C
034A	CB92	04300	RES	2,D
034C	CB93	04310	RES	2,E
034E	CB94	04320	RES	2,H
0350	CB95	04330	RES	2,L
0352	CB9E	04340	RES	3, (HL)
0354	DDCB059E	04350	RES	3, (IX+IND)
0358	FDCB059E	04360	RES	3, (IY+IND)
035C	CB9F	04370	RES	3,A
035E	CB9E	04380	RES	3,B
0360	CB99	04390	RES	3,C
0362	CB9A	04400	RES	3,D
0364	CB9B	04410	RES	3,E
0366	CB9C	04420	RES	3,H
0368	CB9D	04430	RES	3,L
036A	CBA6	04440	RES	4, (HL)
036C	DDCB05A6	04450	RES	4, (IX+IND)
0370	FDCB05A6	04460	RES	4, (IY+IND)
0374	CBA7	04470	RES	4,A
0376	CBA0	04480	RES	4,B
0378	CBA1	04490	RES	4,C
037A	CBA2	04500	RES	4,D
037C	CBA3	04510	RES	4,E
037E	CBA4	04520	RES	4,H
0380	CBA5	04530	RES	4,L

0382	CBAE	04540	RES	5, (HL)
0384	DDCB05AE	04550	RES	5, (IX+IND)
0388	FDCB05AE	04560	RES	5, (IY+IND)
038C	CBAF	04570	RES	5,A
038E	CBAB	04580	RES	5,B
0390	CBA9	04590	RES	5,C
0392	CBAA	04600	RES	5,D
0394	CBAB	04610	RES	5,E
0396	CBAC	04620	RES	5,H
0398	CBAD	04630	RES	5,L
039A	CBB6	04640	RES	6, (HL)
039C	DDCB05B6	04650	RES	6, (IX+IND)
03A0	FDCB05B6	04660	RES	6, (IY+IND)
03A4	CBB7	04670	RES	6,A
03A6	CBB0	04680	RES	6,B
03A8	CBB1	04690	RES	6,C
03AA	CBB2	04700	RES	6,D
03AC	CBB3	04710	RES	6,E
03AE	CBB4	04720	RES	6,H
03B0	CBB5	04730	RES	6,L
03B2	CBBE	04740	RES	7, (HL)
03B4	DDCB05BE	04750	RES	7, (IX+IND)
03B8	FDCB05BE	04760	RES	7, (IY+IND)
03BC	CBBF	04770	RES	7,A
03BE	CBB6	04780	RES	7,B
03C0	CBB9	04790	RES	7,C
03C2	CBBA	04800	RES	7,D
03C4	CBBB	04810	RES	7,E
03C6	CBBC	04820	RES	7,H
03C8	CBBD	04830	RES	7,L
03CA	C9	04840	RET	
03CB	D8	04850	RET	C
03CC	F8	04860	RET	M
03CD	D0	04870	RET	NC
03CE	C0	04880	RET	NZ
03CF	F0	04890	RET	P
03D0	EB	04900	RET	PE
03D1	E0	04910	RET	PO
03D2	C8	04920	RET	Z
03D3	ED4D	04930	RETI	
03D5	ED45	04940	RETJ	
03D7	CB16	04950	RL	(HL)
03D9	DDCB0516	04960	RL	(IX+IND)
03DD	FDCB0516	04970	RL	(IY+IND)
03E1	CB17	04980	RL	A
03E3	CB10	04990	RL	B
03E5	CB11	05000	RL	C
03E7	CB12	05010	RL	D
03E9	CB13	05020	RL	E
03EB	CB14	05030	RL	H
03ED	CB15	05040	RL	L
03EF	17	05050	RLA	
03F0	CB06	05060	RLC	(HL)
03F2	DDCB0506	05070	RLC	(IX+IND)
03F6	FDCB0506	05080	RLC	(IY+IND)
03FA	CB07	05090	RLC	A
03FC	CB00	05100	RLC	B
03FE	CB01	05110	RLC	C
0400	CB02	05120	RLC	D
0402	CB03	05130	RLC	E
0404	CB04	05140	RLC	H
0406	CB05	05150	RLC	L
0408	07	05160	RLCA	
0409	ED6F	05170	RLD	
040E	CB1E	05180	RR	(HL)
040D	DDCB051E	05190	RR	(IX+IND)
0411	FDCB051E	05200	RR	(IY+IND)
0415	CB1F	05210	RR	A

0417	CB1B	05220	RR	B
0419	CB19	05230	RR	C
041B	CB1A	05240	RR	D
041D	CB1B	05250	RR	E
041F	CB1C	05260	RR	H
0421	CB1D	05270	RR	L
0423	1F	05280	RRR	
0424	CB0E	05290	RRC	(HL)
0426	DDCB050E	05300	RRC	(IX+IND)
042A	FDCB050E	05310	RRC	(IY+IND)
042E	CB0F	05320	RRC	A
0430	CB0B	05330	RRC	B
0432	CB09	05340	RRC	C
0434	CB0A	05350	RRC	D
0436	CB0B	05360	RRC	E
043B	CB0C	05370	RRC	H
043A	CB0D	05380	RRC	L
043C	OF	05390	RRCA	
043D	ED67	05400	RRD	
043F	C7	05410	RST	0
0440	D7	05420	RST	10H
0441	DF	05430	RST	18H
0442	E7	05440	RST	20H
0443	EF	05450	RST	28H
0444	F7	05460	RST	30H
0445	FF	05470	RST	38H
0446	CF	05480	RST	B
0447	9E	05490	SBC	A, (HL)
0448	DD9E05	05500	SBC	A, (IX+IND)
044B	FD9E05	05510	SEC	A, (IY+IND)
044E	9F	05520	SBC	A, A
044F	9B	05530	SBC	A, B
0450	99	05540	SBC	A, C
0451	9A	05550	SBC	A, D
0452	9B	05560	SBC	A, E
0453	9C	05570	SBC	A, H
0454	9D	05580	SBC	A, L
0455	DE10	05590	SEC	A, N
0457	ED42	05600	SBC	HL, BC
0459	ED52	05610	SBC	HL, DE
045B	ED62	05620	SBC	HL, HL
045D	ED72	05630	SBC	HL, SP
045F	37	05640	SCF	
0460	CBC6	05650	SET	0, (HL)
0462	DDCB05C6	05660	SET	0, (IX+IND)
0466	FDCB05C6	05670	SET	0, (IY+IND)
046A	CBC7	05680	SET	0, A
046C	CBC0	05690	SET	0, B
046E	CBC1	05700	SET	0, C
0470	CBC2	05710	SET	0, D
0472	CBC3	05720	SET	0, E
0474	CBC4	05730	SET	0, H
0476	CBC5	05740	SET	0, L
047B	CBCE	05750	SET	1, (HL)
047A	DDCB05CE	05760	SET	1, (IX+IND)
047E	FDCB05CE	05770	SET	1, (IY+IND)
0482	CBCF	05780	SET	1, A
0484	CBCB	05790	SET	1, B
0486	CBC9	05800	SET	1, C
048B	CBCA	05810	SET	1, D
048A	CBCB	05820	SET	1, E
048C	CBCC	05830	SET	1, H
048E	CBCE	05840	SET	1, L
0490	CB06	05850	SET	2, (HL)
0492	DDCB05D6	05860	SET	2, (IX+IND)
0496	FDCB05D6	05870	SET	2, (IY+IND)
049A	CB07	05880	SET	2, A
049C	CB06	05890	SET	2, B

049E	CBD1	05900	SET	2, C
04A0	CBD2	05910	SET	2, D
04A2	CBD3	05920	SET	2, E
04A4	CBD4	05930	SET	2, H
04A6	CBD5	05940	SET	2, L
04AB	CBDE	05950	SET	3, (HL)
04AA	DDCB05DE	05960	SET	3, (IX+IND)
04AE	FDCB05DE	05970	SET	3, (IY+IND)
04B2	CBDF	05980	SET	3, A
04B4	CBDB	05990	SET	3, B
04B6	CB09	06000	SET	3, C
04BB	CBDA	06010	SET	3, D
04BA	CBDB	06020	SET	3, E
04BC	CBDC	06030	SET	3, H
04BE	CBDD	06040	SET	3, L
04C0	CBE6	06050	SET	4, (HL)
04C2	DDCB05E6	06060	SET	4, (IX+IND)
04C6	FDCB05E6	06070	SET	4, (IY+IND)
04CA	CBE7	06080	SET	4, A
04CC	CBE0	06090	SET	4, B
04CE	CBE1	06100	SET	4, C
04D0	CBE2	06110	SET	4, D
04D2	CBE3	06120	SET	4, E
04D4	CBE4	06130	SET	4, H
04D6	CBE5	06140	SET	4, L
04D8	CBEE	06150	SET	5, (HL)
04DA	DDCB05EE	06160	SET	5, (IX+IND)
04DE	FDCB05EE	06170	SET	5, (IY+IND)
04E2	CBEF	06180	SET	5, A
04E4	CBE8	06190	SET	5, B
04E6	CBE9	06200	SET	5, C
04E8	CBEA	06210	SET	5, D
04EA	CBEB	06220	SET	5, E
04EC	CBEC	06230	SET	5, H
04EE	CBED	06240	SET	5, L
04F0	CBF6	06250	SET	6, (HL)
04F2	DDCB05F6	06260	SET	6, (IX+IND)
04F6	FDCB05F6	06270	SET	6, (IY+IND)
04FA	CBF7	06280	SET	6, A
04FC	CBF0	06290	SET	6, B
04FE	CBF1	06300	SET	6, C
0500	CBF2	06310	SET	6, D
0502	CBF3	06320	SET	6, E
0504	CBF4	06330	SET	6, H
0506	CBF5	06340	SET	6, L
0508	CBFE	06350	SET	7, (HL)
050A	DDCB05FE	06360	SET	7, (IX+IND)
050E	FDCB05FE	06370	SET	7, (IY+IND)
0512	CBFF	06380	SET	7, A
0514	CBF8	06390	SET	7, B
0516	CBF9	06400	SET	7, C
0518	CBFA	06410	SET	7, D
051A	CBFB	06420	SET	7, E
051C	CBFC	06430	SET	7, H
051E	CBFD	06440	SET	7, L
0520	CB26	06450	SLA	(HL)
0522	DDCB0526	06460	SLA	(IX+IND)
0526	FDCB0526	06470	SLA	(IY+IND)
052A	CB27	06480	SLA	A
052C	CB20	06490	SLA	B
052E	CB21	06500	SLA	C
0530	CB22	06510	SLA	D
0532	CB23	06520	SLA	E
0534	CB24	06530	SLA	H
0536	CB25	06540	SLA	L
0538	CB2E	06550	SRA	(HL)
053A	DDCB052E	06560	SRA	(IX+IND)
053E	FDCB052E	06570	SRA	(IY+IND)

0542	CB2F	06580	SRA	A	
0544	CB28	06590	SRA	B	
0546	CB29	06600	SRA	C	
0548	CB2A	06610	SRA	D	
054A	CB2B	06620	SRA	E	
054C	CB2C	06630	SRA	H	
054E	CB2D	06640	SRA	L	
0550	CB3E	06650	SRL	(HL)	
0552	DDCB033E	06660	SRL	(IX+IND)	
0556	FDCB033E	06670	SRL	(IY+IND)	
055A	CB3F	06680	SRL	A	
055C	CB3B	06690	SRL	B	
055E	CB39	06700	SRL	C	
0560	CB3A	06710	SRL	D	
0562	CB3B	06720	SRL	E	
0564	CB3C	06730	SRL	H	
0566	CB3D	06740	SRL	L	
0568	96	06750	SUB	(HL)	
0569	DD9605	06760	SUB	(IX+IND)	
056C	FD9605	06770	SUB	(IY+IND)	
056F	97	06780	SUB	A	
0570	90	06790	SUB	B	
0571	91	06800	SUB	C	
0572	92	06810	SUB	D	
0573	93	06820	SUB	E	
0574	94	06830	SUB	H	
0575	95	06840	SUB	L	
0576	D610	06850	SUB	N	
0578	AE	06860	XOR	(HL)	
0579	DDAE05	06870	XOR	(IX+IND)	
057C	FDAE05	06880	XOR	(IY+IND)	
057F	AF	06890	XOR	A	
0580	AB	06900	XOR	B	
0581	A9	06910	XOR	C	
0582	AA	06920	XOR	D	
0583	AB	06930	XOR	E	
0584	AC	06940	XOR	H	
0585	AD	06950	XOR	L	
0586	EE10	06960	XOR	N	
FFEE		06970	EQU	OFFEEH	
0030		06980	EQU	30H	; Número de 16 bits
0005		06990	EQU	5	; Distancia relativa
0010		07000	EQU		; Índice
0000		07010	EQU	10H	; Número de 8 bits
			END		

## Otros libros sobre MICROINFORMATICA

- 
- |             |                                                                     |
|-------------|---------------------------------------------------------------------|
| C. Prigmore | <b>MICROSOFT BASIC</b><br>Curso de autoenseñanza para principiantes |
|-------------|---------------------------------------------------------------------|
- 
- |            |                                                                  |
|------------|------------------------------------------------------------------|
| G. Ladevie | <b>La gestión con BASIC</b><br>Comercio y pequeña empresa        |
| G. Guérin  | <b>Microinformática de gestión</b><br>Alternativas y utilización |
- 
- |                        |                                                                            |
|------------------------|----------------------------------------------------------------------------|
| A.P. Mullan            | <b>El ordenador en la Educación Básica</b><br>Problemática y metodología   |
| D. Daines              | <b>Las bases de datos en la Educación Básica</b><br>Utilización y ejemplos |
| G.W. Orwig/W.S. Hodges | <b>Programas educativos para su ordenador personal</b>                     |
- 
- |             |                                                                       |
|-------------|-----------------------------------------------------------------------|
| P. Pellier  | <b>Lenguaje máquina del ZX Spectrum</b><br>Subrutinas y trucos        |
| T. Hartnell | <b>Juegos dinámicos para el ZX Spectrum</b>                           |
| R.G. Hurley | <b>Los Micro Drives del ZX Spectrum</b><br>Utilización y aplicaciones |
- 
- |             |                                                |
|-------------|------------------------------------------------|
| I. Sinclair | <b>Introducción al Commodore 64</b>            |
| I. Sinclair | <b>Lenguaje máquina del Commodore 64</b>       |
| S. Money    | <b>Gráficos y sonidos para el Commodore 64</b> |
| O. Bishop   | <b>Juegos para el Commodore 64</b>             |
- 
- |                             |                                                                                  |
|-----------------------------|----------------------------------------------------------------------------------|
| B. Lloyd                    | <b>Introducción al Dragon</b>                                                    |
| D. Lawrence                 | <b>Programas prácticos para el Dragon</b>                                        |
| K.S. Brain                  | <b>Gráficos y sonidos para el Dragon</b><br>Incluye subrutinas en código máquina |
| K.S. Brain                  | <b>Inteligencia artificial en el Dragon</b>                                      |
| I. Sinclair                 | <b>Lenguaje máquina del Dragon</b>                                               |
| M. James/S.M. Gee/K. Ewbank | <b>Juegos para el Dragon</b>                                                     |
| V. Apps                     | <b>40 juegos educativos para el Dragon</b>                                       |



## 54LS/74LS245

### OCTAL BUS TRANSCEIVER

(With 3-State Outputs)

**DESCRIPTION** — The 'LS245 is an octal bus transmitter/receiver designed for 8-line asynchronous 2-way data communication between data busses. Direction input (DR) controls transmission of data from bus A to bus B or bus B to bus A depending upon its logic level. The Enable input ( $\bar{E}$ ) can be used to isolate the busses.

- HYSTERESIS INPUTS TO IMPROVE NOISE IMMUNITY
- 2-WAY ASYNCHRONOUS DATA BUS COMMUNICATION
- INPUT DIODES LIMIT HIGH SPEED TERMINATION EFFECTS
- FULLY TTL AND CMOS COMPATIBLE

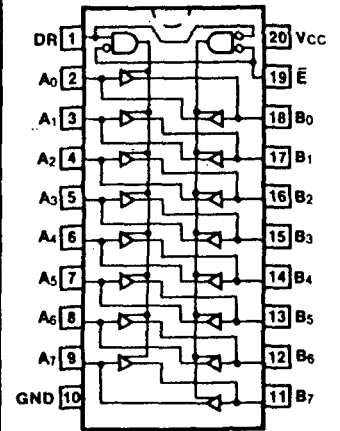
**ORDERING CODE:** See Section 9

PKGS	PIN OUT	COMMERCIAL GRADE	MILITARY GRADE	PKG TYPE
		$V_{CC} = +5.0\text{ V} \pm 5\%$ , $T_A = 0^\circ\text{C to } +70^\circ\text{C}$	$V_{CC} = +5.0\text{ V} \pm 10\%$ , $T_A = -55^\circ\text{C to } +125^\circ\text{C}$	
Plastic DIP (P)	A	74LS245PC		9Z
Ceramic DIP (D)	A	74LS245DC	54LS245DM	4E
Flatpak (F)	A	74LS245FC	54LS245FM	4F

**INPUT LOADING/FAN-OUT:** See Section 3 for U.L. definitions

PINS	54/74LS (U.L.) HIGH/LOW
Inputs	0.5/0.125
Outputs	75/15 (7.5)

#### CONNECTION DIAGRAM PINOUT A



#### TRUTH TABLE

INPUTS		OUTPUT
$\bar{E}$	DR	
L	L	Bus B Data to Bus A
L	H	Bus A Data to Bus B
H	X	Isolation

H = HIGH Voltage Level  
L = LOW Voltage Level  
X = Immaterial

## DC CHARACTERISTICS OVER OPERATING TEMPERATURE RANGE (unless otherwise specified)

SYMBOL	PARAMETER		54/74LS		UNITS	CONDITIONS
			Min	Max		
VOH	Output HIGH Voltage	XM	2.0		V	V <sub>CC</sub> = Min V <sub>IN</sub> = V <sub>IH</sub> or V <sub>IL</sub> per Truth Table
		XC	2.0			
VOH	Output HIGH Voltage		2.4		V	I <sub>OH</sub> = -3.0 mA, V <sub>CC</sub> = Min V <sub>IN</sub> = V <sub>IH</sub> or V <sub>IL</sub> per Truth Table
V <sub>T+</sub> - V <sub>T-</sub>	Hysteresis Voltage		0.2		V	V <sub>CC</sub> = Min
I <sub>OS</sub>	Output Short Circuit Current		-40	-225	mA	V <sub>CC</sub> = Max, V <sub>OUT</sub> = 0 V
I <sub>CC</sub>	Power Supply Current	HIGH		70	mA	V <sub>CC</sub> = Max
		LOW		90		
		OFF		95		

AC CHARACTERISTICS: V<sub>CC</sub> = +5.0 V, T<sub>A</sub> = +25°C (See Section 3 for waveforms and load configurations)

SYMBOL	PARAMETER		54/74LS		UNITS	CONDITIONS
			C <sub>L</sub> = 45 pF			
			Min	Max		
t <sub>PLH</sub> t <sub>PHL</sub>	Propagation Delay Data to Output		18 18		ns	Figs. 3-1, 3-5
t <sub>PZH</sub> t <sub>PZL</sub>	Output Enable Time		25 30		ns	Figs. 3-3, 3-11, 3-12 R <sub>L</sub> = 667 Ω
t <sub>PLZ</sub> t <sub>PHZ</sub>	Output Disable Time		25 18		ns	Figs. 3-3, 3-11, 3-12 R <sub>L</sub> = 667 Ω, C <sub>L</sub> = 5 pF

**RS**  
**data**

# 8 Bit D to A/A to D converter i.c.

The ZN425E is an 8 bit dual mode analogue to digital/digital to analogue converter. It contains an 8 bit D to A converter using an advance design of R-2R ladder network and an array of precision bipolar switches plus an 8 bit binary counter and a 2.5 volt precision voltage reference all on a single monolithic chip.

The special design of ladder network results in full 8 bit accuracy using normal diffused resistors.

The use of the on-chip reference voltage is pin optional to retain flexibility. An external fixed or varying reference may therefore be substituted.

By including on the chip an 8 bit binary counter, analogue to digital conversion can be obtained simply by adding an external comparator (531) and clock inhibit gating (7400).

By simply clocking the counter the ZN425E can be used as a self-contained precision staircase ramp generator.

A logic input select switch is incorporated which determines whether the precision switches accept the outputs from the binary counter or external digital inputs depending upon whether the control signal is respectively high or low.

## Features

- Dual mode, digital to analogue/analogue to digital.
- On-chip precision voltage reference.
- Includes 8 bit binary counter.
- Will function as precision staircase ramp generator.
- TTL and CMOS compatible.
- Direct voltage output.
- 16 pin D.I.L. encapsulation.

## Pin connections (Top view)

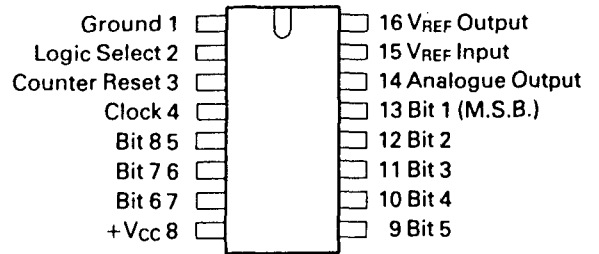
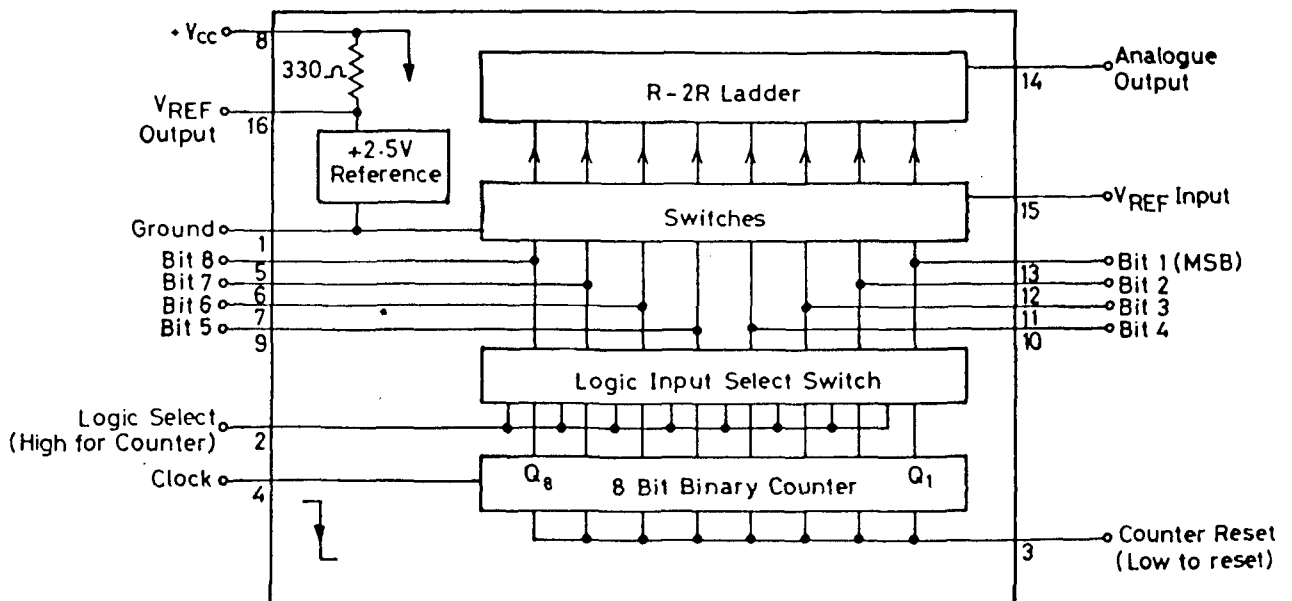


Figure 1: Block diagram



**Absolute maximum ratings**

Supply voltage  $V_{CC}$  \_\_\_\_\_ +7.0 volts  
 Max. voltage, logic and  $V_{REF}$  inputs \_\_\_\_\_ +5.5 volts

Operating temperature range \_\_\_\_\_ 0 to +70°C  
 Storage temperature range \_\_\_\_\_ -55 to +125°C

**Characteristics** At  $T_{amb} = 25^\circ\text{C}$  and  $V_{CC} = +5$  volts unless otherwise specified.

Parameter	Symbol	Conditions	Min.	Typ.	Max.	Units
Supply voltage	$V_{CC}$		4.5	—	5.5	volts
Supply current	$I_s$		—	30	40	mA
High level input voltage	$V_{IH}$	See notes 1 and 2	2.0	—	—	volts
Low level input voltage	$V_{IL}$		—	—	0.7	volts
High level input current	$I_{IH}$	$V_{CC} = \text{max.}, V_I = 2.4\text{V}$ $V_{CC} = \text{max.}, V_I = 5.5\text{V}$	— —	— —	10 100	$\mu\text{A}$ $\mu\text{A}$
Low level input current	$I_{IL}$	$V_{CC} = \text{max.}, V_I = 0.3\text{V}$	—	—	-0.68	mA
High level output current	$I_{OH}$		—	—	-40	$\mu\text{A}$
Low level output current	$I_{OL}$		—	—	1.6	mA
High level output voltage	$V_{OH}$	$V_{CC} = \text{min.}, Q = 1,$ $I_{load} = -40\mu\text{A}$	2.4	—	—	volts
Low level output voltage	$V_{OL}$	$V_{CC} = \text{min.}, Q = 0,$ $I_{load} = 1.6\text{mA}$	—	—	0.4	volts

**8 Bit D to A converter**

Parameter	Symbol	Conditions	Min.	Typ.	Max.	Units
Linearity error			—	—	$\pm 0.5$	L.S.B.
Settling time		1 L.S.B. step	—	1.0	—	$\mu\text{s}$
Settling time to 0.5 L.S.B.		All bits ON to OFF or OFF to ON	—	2.0	—	$\mu\text{s}$
Offset voltage	$V_{OS}$	All bits OFF	—	3.0	—	mV
$V_{OS}$ temperature coefficient			—	5	—	$\mu\text{V}/^\circ\text{C}$
F.S.R. temperature coefficient		Ext. $V_{REF} = 2.5\text{V}$	—	3	—	ppm/ $^\circ\text{C}$
Linearity error temperature coefficient		Relative to F.S.R	—	7.5	—	ppm/ $^\circ\text{C}$
Analogue output resistance	$R_o$		—	10	—	k ohm
External reference voltage			0	—	3.0	volts

**Internal voltage reference**

Parameter	Symbol	Conditions	Min.	Typ.	Max.	Units
Output voltage	$V_{REF}$	$I = 7.5\text{mA}$ (internal)	—	2.55	—	volts
Slope resistance	$R_s$	$I = 7.5\text{mA}$ (internal)	—	2	—	ohms
$V_{REF}$ temperature coefficient		$I = 7.5\text{mA}$ (internal)	—	40	—	ppm/ $^\circ\text{C}$

**Notes:**

- The logic select pin (2) must be held low when the bit pins (5, 6, 7, 9, 10, 11, 12 and 13) are driven externally.
- To obtain counter outputs on bit pins the logic

select pin (2) should be taken to  $+V_{CC}$  via a  $1\text{ k}\Omega$  resistor.

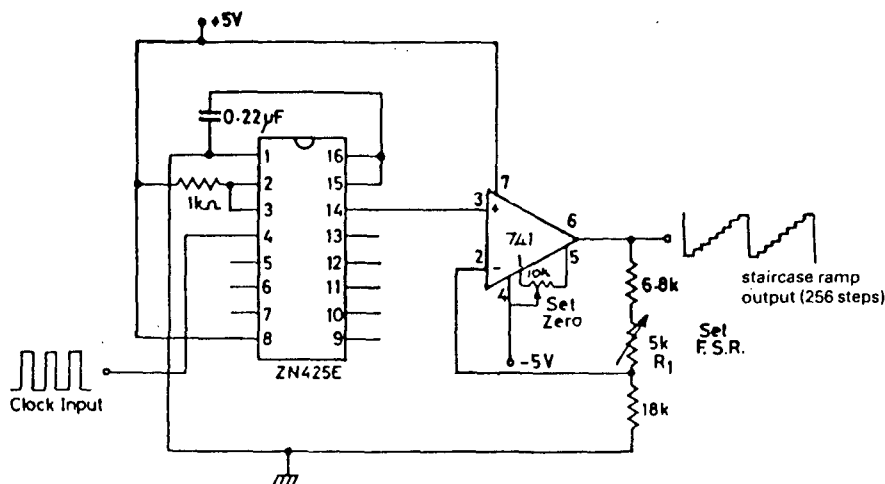
- The internal reference requires a  $0.22\ \mu\text{F}$  stabilising capacitor between pins 1 and 16.

**RS data****8 Bit D to A/A to D converter i.c.****Precision staircase ramp generator**

The inclusion of an 8 bit binary counter on the chip gives the ZN425E a useful staircase ramp generator function. The circuit, Fig 4 uses the same buffer stage as the D to A converter. The calibration procedure is also the same. Holding pin 2 low will set all bits to ON and if RESET is taken low with pin 2

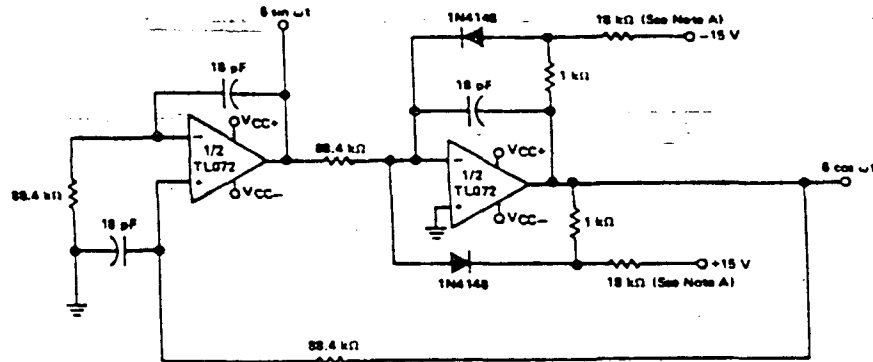
high all the bits are turned to OFF. If the end voltages of the ramp are not required to be set accurately then the buffer stage could be omitted and the voltage ramp will appear directly at pin 14. (N.B. Resetting of the output may take place at any point of the waveform by taking pin 3 [reset] low.)

Figure 4: Precision staircase ramp generator



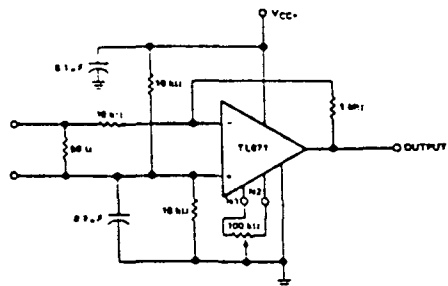
**TYPES TL070, TL070A, TL071, TL071A, TL071B,  
TL072, TL072A, TL072B, TL074, TL074A, TL074B, TL075  
LOW-NOISE JFET-INPUT OPERATIONAL AMPLIFIERS**

**TYPICAL APPLICATION DATA**

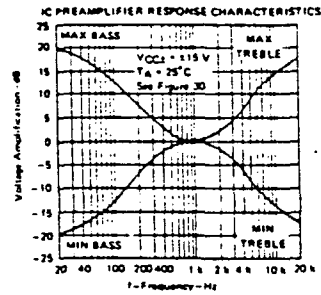


Note A: These resistor values may be adjusted for a symmetrical output.

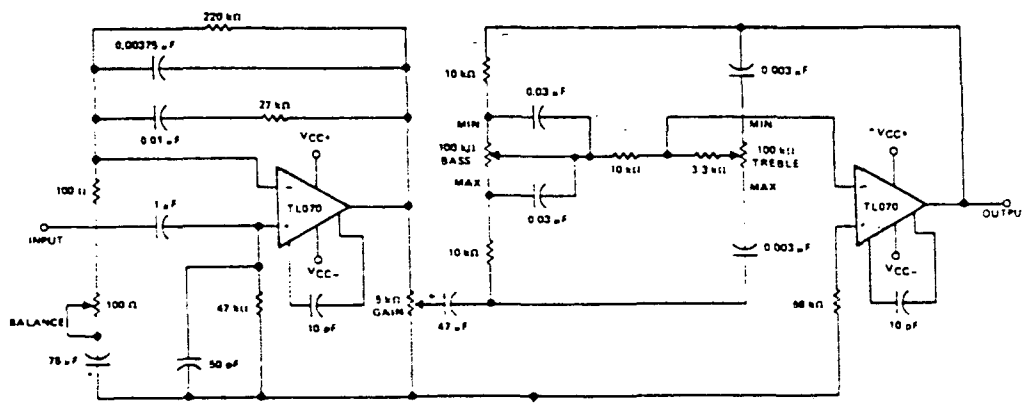
**FIGURE 29—100-KHz QUADRATURE OSCILLATOR**



**FIGURE 30—AC AMPLIFIER**



**FIGURE 31**



**FIGURE 32—IC PREAMPLIFIER**

**LINEAR  
INTEGRATED  
CIRCUITS**

**TYPES TL080 THRU TL085, TL080A THRU TL084A,  
TL081B, TL082B, TL084B  
JFET-INPUT OPERATIONAL AMPLIFIERS**

BULLETIN NO. DLS 12464, FEBRUARY 1977—REVISED OCTOBER 1979

24 DEVICES COVER COMMERCIAL, INDUSTRIAL, AND MILITARY TEMPERATURE RANGES

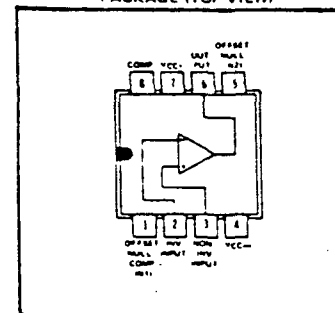
- Low Power Consumption
- Wide Common-Mode and Differential Voltage Ranges
- Low Input Bias and Offset Currents
- Output Short-Circuit Protection
- High Input Impedance . . . JFET-Input Stage
- Internal Frequency Compensation (Except TL080, TL080A)
- Latch-Up-Free Operation
- High Slew Rate . . . 13 V/μs Typ

**description**

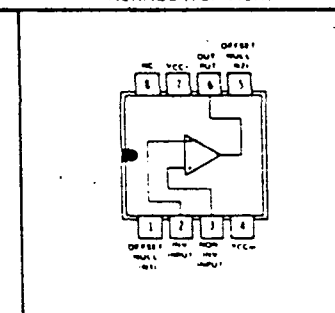
The TL081 JFET-input operational amplifier family is designed to offer a wider selection than any previously developed operational amplifier family. Each of these JFET-input operational amplifiers incorporates well-matched, high-voltage JFET and bipolar transistors in a monolithic integrated circuit. The devices feature high slew rates, low input bias and offset currents, and low offset voltage temperature coefficient. Offset adjustment and external compensation options are available within the TL081 Family.

Device types with an "M" suffix are characterized for operation over the full military temperature range of -55°C to 125°C, those with an "I" suffix are characterized for operation from -25°C to 85°C, and those with a "C" suffix are characterized for operation from 0°C to 70°C.

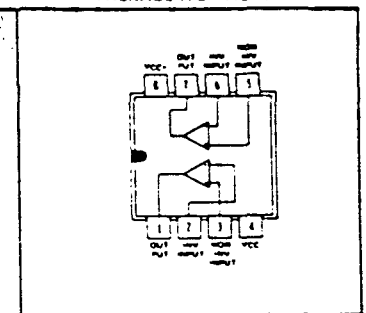
**TL080, TL080A  
JG OR P DUAL-IN-LINE  
PACKAGE (TOP VIEW)**



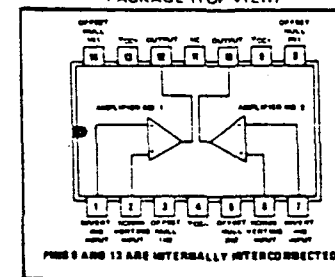
**TL081, TL081A, TL081B  
JG OR P DUAL-IN-LINE  
PACKAGE (TOP VIEW)**



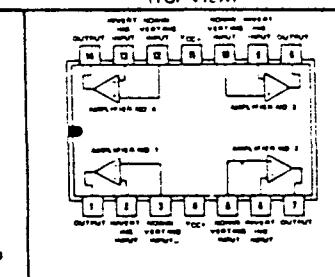
**TL082, TL082A, TL082B  
JG OR P DUAL-IN-LINE  
PACKAGE (TOP VIEW)**



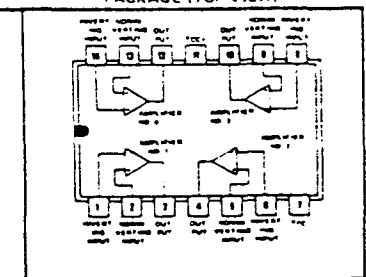
**TL083, TL083A  
J OR N DUAL-IN-LINE  
PACKAGE (TOP VIEW)**



**TL084, TL084A, TL084B  
J OR N DUAL-IN-LINE  
OR W FLAT PACKAGE  
(TOP VIEW)**



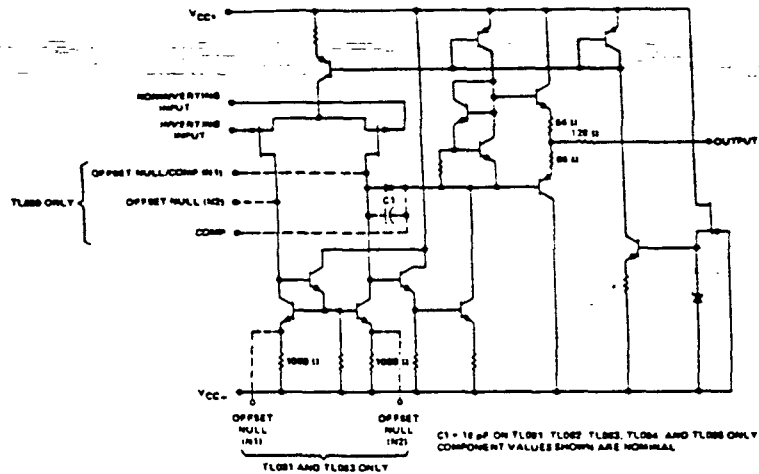
**TL085  
N DUAL-IN-LINE  
PACKAGE (TOP VIEW)**



NC—No internal connection

# TYPES TL080 THRU TL085, TL080A THRU TL084A, TL081B, TL082B, TL084B JFET-INPUT OPERATIONAL AMPLIFIERS

schematic (each amplifier)



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

	TL08_M	TL08_I	TL08_C TL08_AC TL08_BC	UNIT
Supply voltage, $V_{CC+}$ (see Note 1)	18	18	18	V
Supply voltage, $V_{CC-}$ (see Note 1)	-18	-18	-18	V
Differential input voltage (see Note 2)	$\pm 30$	$\pm 30$	$\pm 30$	V
Input voltage (see Notes 1 and 3)	$\pm 15$	$\pm 15$	$\pm 15$	V
Duration of output short circuit (see Note 4)	Unlimited	Unlimited	Unlimited	
Continuous total dissipation at (or below) 25°C free-air temperature (see Note 5)	680	680	680	mW
Operating free-air temperature range	-55 to 125	-25 to 85	0 to 70	°C
Storage temperature range	-65 to 150	-65 to 150	-65 to 150	°C
Lead temperature 1/16 inch (1.6 mm) from case for 60 seconds: J, JG, or W package	300	300	300	°C
Lead temperature 1/16 inch (1.6 mm) from case for 10 seconds: N or P package	260	260	260	°C

- NOTES: 1. All voltage values, except differential voltages, are with respect to the midpoint between  $V_{CC+}$  and  $V_{CC-}$ .  
 2. Differential voltages are at the noninverting input terminal with respect to the inverting input terminal.  
 3. The magnitude of the input voltage must never exceed the magnitude of the supply voltage or 15 volts, whichever is less.  
 4. The output may be shorted to ground or to either supply. Temperature and/or supply voltages must be limited to ensure that the dissipation rating is not exceeded.  
 5. For operation above 25°C free-air temperature, refer to Dissipation Derating Table. In the J and JG packages, TL08\_M chips are alloy-mounted; TL08\_I, TL08\_C, TL08\_AC, and TL08\_BC chips are glass-mounted.

DISSIPATION DERATING TABLE

PACKAGE	POWER RATING	DERATING FACTOR	ABOVE $T_A$
J (Alloy-Mounted Chip)	680 mW	11.0 mW/°C	68°C
JG (Glass-Mounted Chip)	680 mW	6.2 mW/°C	67°C
JG (Alloy-Mounted Chip)	680 mW	8.4 mW/°C	69°C
JG (Glass-Mounted Chip)	680 mW	6.6 mW/°C	47°C
N	680 mW	9.2 mW/°C	78°C
P	680 mW	8.0 mW/°C	65°C
W	680 mW	8.0 mW/°C	65°C

Also see Dissipation Derating Curves, Section 2

DEVICE TYPES SUFFIX VERSIONS, AND PACKAGES

	TL080	TL081	TL082	TL083	TL084	TL085
TL08_M	JG	JG	JG	J	J, W	*
TL08_I	JG, P	JG, P	JG, P	J, N	J, N	*
TL08_C	JG, P	JG, P	JG, P	J, N	J, N	N
TL08_AC	JG, P	JG, P	JG, P	J, N	J, N	*
TL08_BC	JG, P	JG, P	JG, P	J, N	J, N	*

\*These combinations are not defined by this data sheet.

# TYPES TL080 THRU TL085, TL080A THRU TL084A, TL081B, TL082B, TL084B JFET-INPUT OPERATIONAL AMPLIFIERS

electrical characteristics,  $V_{CC+} = \pm 15$  V

PARAMETER	TEST CONDITIONS†	TL08_M			TL08_I			TL08_C TL08_AC TL08_BC			UNIT
		MIN	TYP	MAX	MIN	TYP	MAX	MIN	TYP	MAX	
$V_{IO}$ Input offset voltage	$R_S = 50 \Omega$ , $T_A = 25^\circ\text{C}$	'80, '81, '82, '83, '85‡									mV
		TL084									
		TL08_A									
		'81B, '82B, '84B									
$V_{IO}$ Input offset voltage	$R_S = 50 \Omega$ , $T_A = \text{full range}$	'80, '81, '82, '83, '85‡									mV
		TL084									
		TL08_A									
		'81B, '82B, '84B									
$\alpha V_{IO}$ Temperature coefficient of input offset voltage	$R_S = 50 \Omega$ , $T_A = \text{full range}$	10			10			10			$\mu\text{V}/^\circ\text{C}$
$I_{IO}$ Input offset current §	$T_A = 25^\circ\text{C}$	TL08_:									pA
		TL08_A									
		'81B, '82B, '84B									
		TL08_:									
$I_{IO}$ Input offset current §	$T_A = \text{full range}$	TL08_:									nA
		TL08_A									
		'81B, '82B, '84B									
		TL08_:									
$I_{IB}$ Input bias current §	$T_A = 25^\circ\text{C}$	TL08_:									pA
		TL08_A									
		'81B, '82B, '84B									
		TL08_:									
$I_{IB}$ Input bias current §	$T_A = \text{full range}$	TL08_:									nA
		TL08_A									
		'81B, '82B, '84B									
		TL08_:									
$V_{ICR}$ Common-mode input voltage range	$T_A = 25^\circ\text{C}$	$\pm 11$ $\pm 12$			$\pm 11$ $\pm 12$			$\pm 10$ $\pm 11$			V
		TL08_:									
		TL08_A									
		'81B, '82B, '84B									
$V_{OPP}$ Maximum peak-to-peak output voltage swing	$T_A = 25^\circ\text{C}$ , $R_L > 10 \text{ k}\Omega$	$R_L > 2 \text{ k}\Omega$ , $V_O = \pm 10 \text{ V}$ , $T_A = 25^\circ\text{C}$									V
		$R_L > 2 \text{ k}\Omega$ , $V_O = \pm 10 \text{ V}$ , $T_A = \text{full range}$									
		TL08_:									
		'81B, '82B, '84B									
$A_{VD}$ Large-signal differential voltage amplification	$R_L > 2 \text{ k}\Omega$ , $V_O = \pm 10 \text{ V}$ , $T_A = \text{full range}$	TL08_:									V/mV
		TL08_A									
		'81B, '82B, '84B									
		TL08_:									
$B_1$ Unity-gain bandwidth	$T_A = 25^\circ\text{C}$	3			3			3			MHz
$r_i$ Input resistance	$T_A = 25^\circ\text{C}$	$10^{11}$			$10^{11}$			$10^{11}$			$\Omega$
CMRR Common-mode rejection ratio	$R_S > 10 \text{ k}\Omega$ , $T_A = 25^\circ\text{C}$	TL08_:									dB
		TL08_A									
		'81B, '82B, '84B									
		TL08_:									
KSVR Supply voltage rejection ratio ( $\Delta V_{CC+}/\Delta V_{IO}$ )	$R_S > 10 \text{ k}\Omega$ , $T_A = 25^\circ\text{C}$	TL08_:									dB
		TL08_A									
		'81B, '82B, '84B									
		TL08_:									
$I_{CC}$ Supply current (per amplifier)	No load, $T_A = 25^\circ\text{C}$	1.4 2.8			1.4 2.8			1.4 2.8			mA
$V_{01}/V_{02}$ Channel separation	$A_{VD} = 100$ , $T_A = 25^\circ\text{C}$	120			120			120			dB

† All characteristics are specified under open-loop conditions unless otherwise noted. Full range for  $T_A$  is -55°C to 125°C for TL08\_M, -25°C to 85°C for TL08\_I, and 0°C to 70°C for TL08\_C, TL08\_AC, and TL08\_BC.

‡ Types TL085I and TL085M are not defined by this data sheet.

§ Input bias currents of a JFET-input operational amplifier are normal junction reverse currents, which are temperature sensitive as shown in Figure 18. Pulse techniques must be used that will maintain the junction temperature as close to the ambient temperature as is possible.

# TYPES TL080 THRU TL085, TL080A THRU TL084A, TL081B, TL082B, TL084B

## JFET-INPUT OPERATIONAL AMPLIFIERS

operating characteristics,  $V_{CC\pm} = \pm 15$  V,  $T_A = 25^\circ$  C

PARAMETER	TEST CONDITIONS	TL08_M			ALL OTHERS			UNIT
		MIN	TYP	MAX	MIN	TYP	MAX	
SR	Slew rate at unity gain $V_I = 10$ V, $C_L = 100$ pF, See Figure 1	8	13		13			V/ $\mu$ s
$t_r$	Rise time $V_I = 20$ mV, $R_L = 2$ k $\Omega$ , See Figure 1		0.1		0.1			$\mu$ s
	Overshoot factor $C_L = 100$ pF, $f = 1$ kHz		10%		10%			
$V_n$	Equivalent input noise voltage $R_g = 100$ $\Omega$ , $f = 1$ kHz		25		25			nV/ $\sqrt{Hz}$

### PARAMETER MEASUREMENT INFORMATION

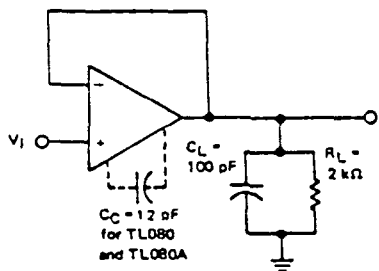


FIGURE 1—UNITY-GAIN AMPLIFIER

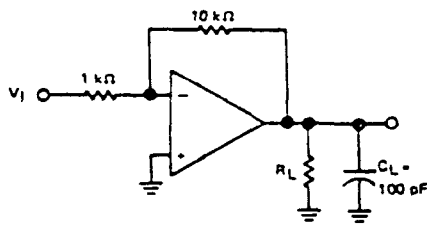


FIGURE 2—GAIN-OF-10 INVERTING AMPLIFIER

### INPUT OFFSET VOLTAGE NULL CIRCUITS

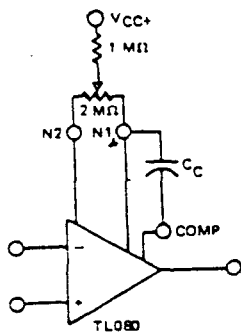


FIGURE 3

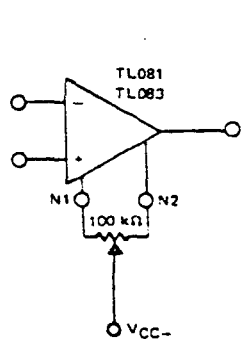
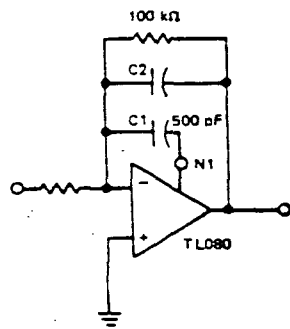


FIGURE 4



FEED-FORWARD COMPENSATION

FIGURE 5

# TYPES TL080 THRU TL085, TL080A THRU TL084A, TL081B, TL082B, TL084B

## JFET-INPUT OPERATIONAL AMPLIFIERS

### TYPICAL CHARACTERISTICS

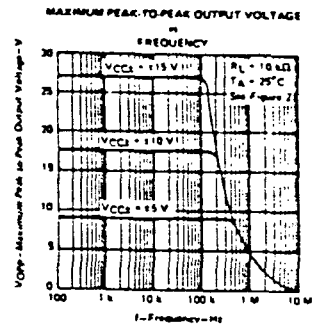


FIGURE 6

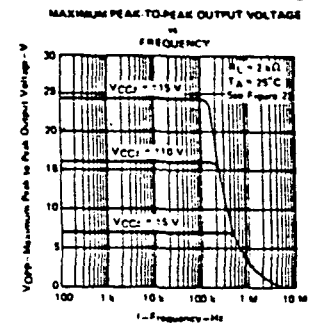


FIGURE 7

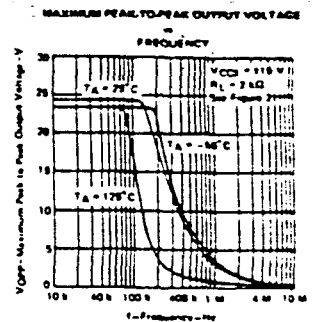


FIGURE 8

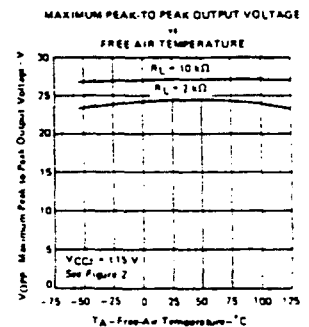


FIGURE 9

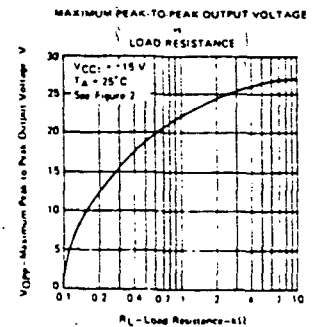


FIGURE 10

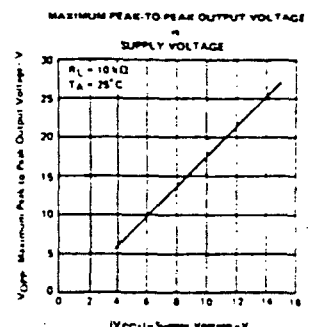


FIGURE 11

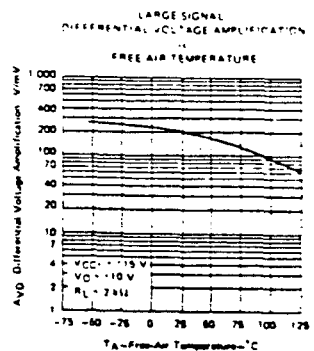


FIGURE 12

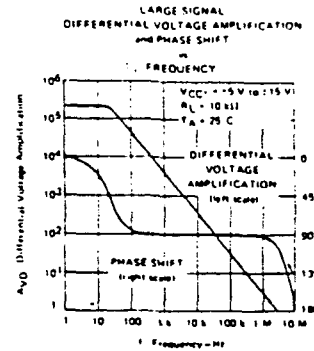


FIGURE 13

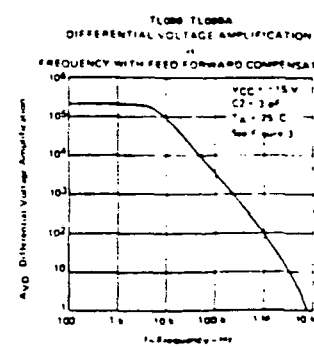


FIGURE 14

† Data at high and low temperatures are applicable only within the rated operating free-air temperature ranges of the various devices. A 12-01



# TYPES TL080 THRU TL085, TL080A THRU TL084A, TL081B, TL082B, TL084B JFET-INPUT OPERATIONAL AMPLIFIERS

## TYPICAL CHARACTERISTICS†

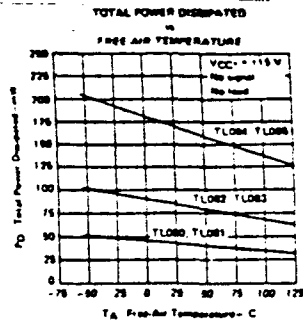


FIGURE 15

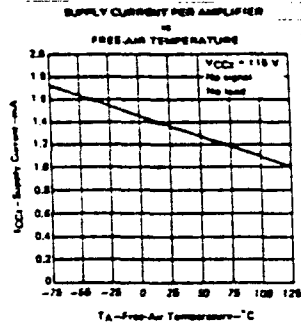


FIGURE 16

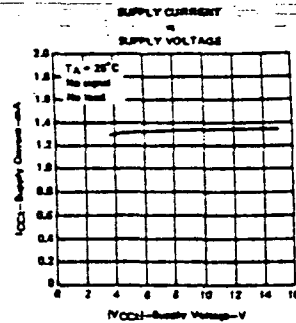


FIGURE 17

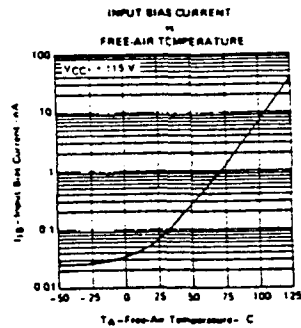


FIGURE 18

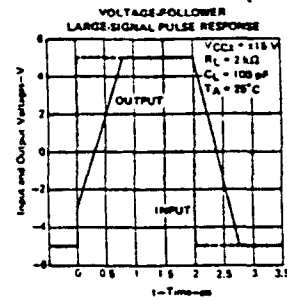


FIGURE 19

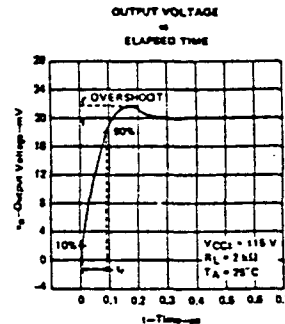


FIGURE 20

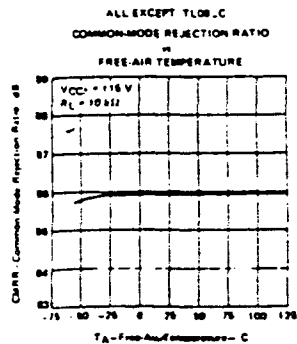


FIGURE 21

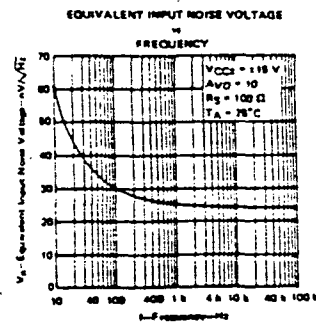


FIGURE 22

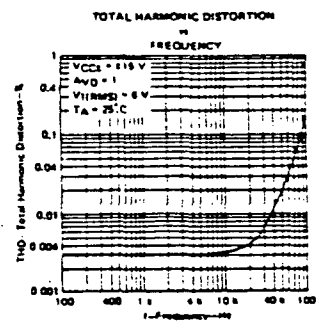


FIGURE 23

# TYPES TL080 THRU TL085, TL080A THRU TL084A, TL081B, TL082B, TL084B JFET-INPUT OPERATIONAL AMPLIFIERS

## TYPICAL APPLICATION DATA

0.5-Hz SQUARE-WAVE OSCILLATOR

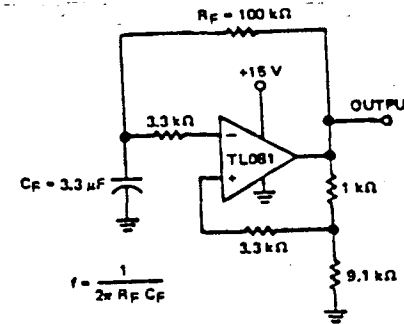


FIGURE 24—0.5-Hz SQUARE-WAVE OSCILLATOR

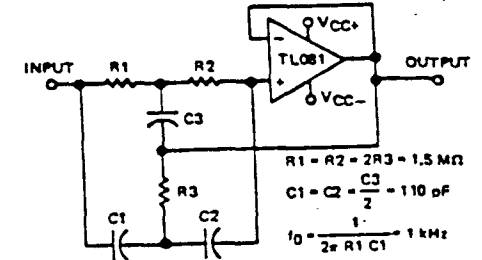
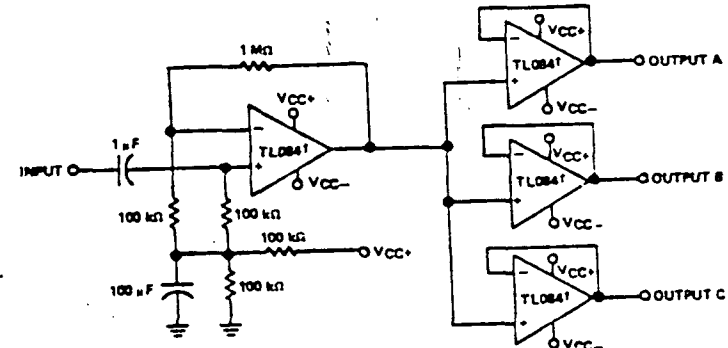
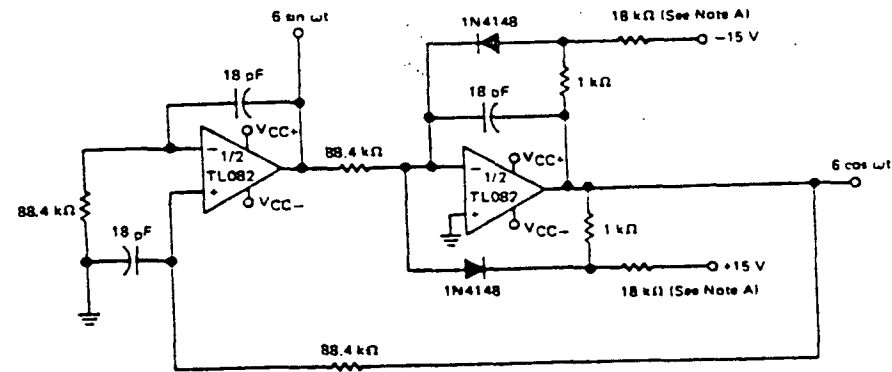


FIGURE 25—HIGH-Q NOTCH FILTER



† or TL085 FIGURE 26—AUDIO DISTRIBUTION AMPLIFIER



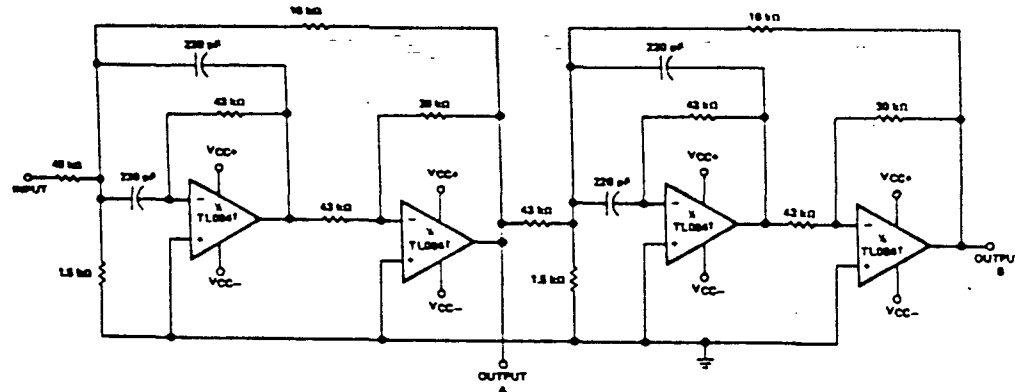
Note A: These resistor values may be adjusted for a symmetrical output.

FIGURE 27—100-kHz QUADRATURE OSCILLATOR

† Data at high and low temperatures are applicable only within the rated operating free-air temperature ranges of the various devices. A 12-μF

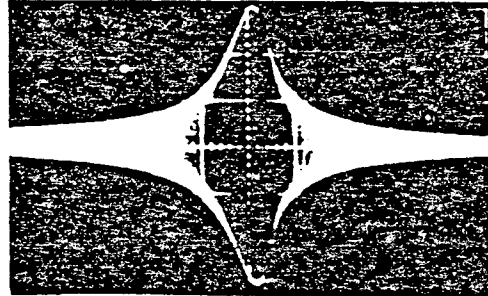
# TYPES TL080 THRU TL085, TL080A THRU TL084A, TL081B, TL082B, TL084B JFET-INPUT OPERATIONAL AMPLIFIERS

## TYPICAL APPLICATION DATA



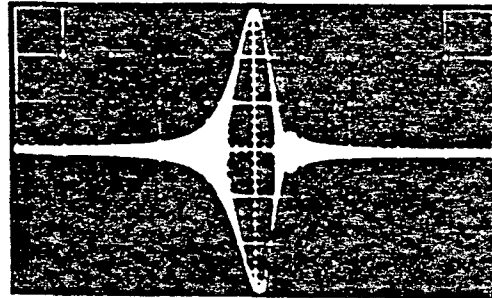
† or TL085

OUTPUT A



2 kHz/div  
SECOND-ORDER BANDPASS FILTER  
 $f_0 = 100 \text{ kHz}$ ,  $Q = 30$ , GAIN = 4

OUTPUT B



2 kHz/div  
CASCADED BANDPASS FILTER  
 $f_0 = 100 \text{ kHz}$ ,  $Q = 69$ , GAIN = 18

FIGURE 28—POSITIVE-FEEDBACK BANDPASS FILTER

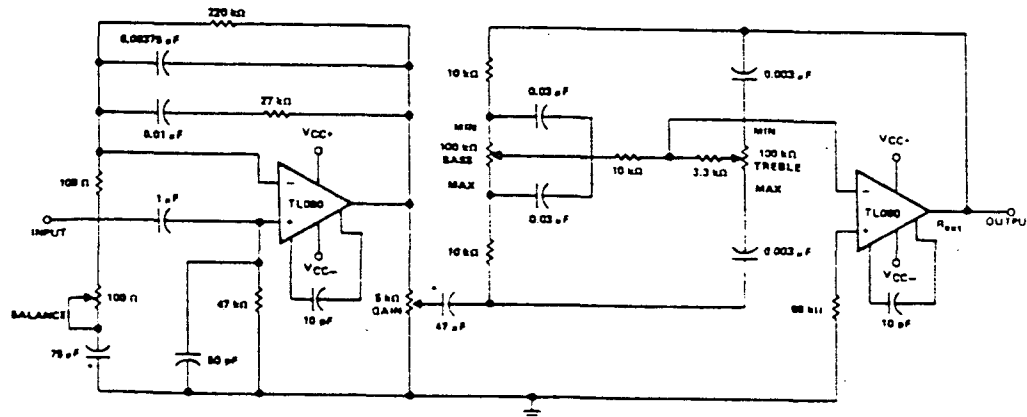


FIGURE 29—IC PREAMPLIFIER

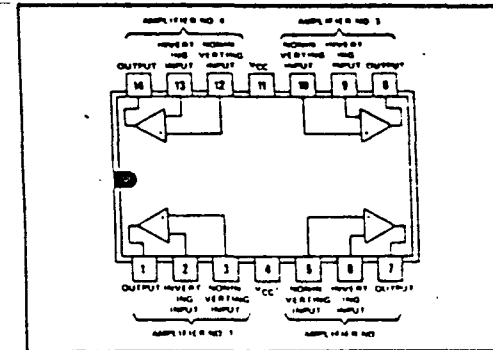
# LINEAR INTEGRATED CIRCUITS

# TYPES TL094M, TL094I, TL094C QUADRUPLER JFET-INPUT OPERATIONAL AMPLIFIERS

BULLETIN NO. OL 5 12754 JUNE 1980

- Wide Range of Supply Voltages  
Single Supply ... 3 V to 36 V  
or Dual Supplies
- Class AB Output Stage
- High-Impedance N-Channel-JFET  
Input Stage ...  $10^{12} \Omega$  typical
- Internal Frequency Compensation
- Short-Circuit Protection
- Input Common-Mode Range Includes  $V_{CC-}$
- Low Input Offset Current ... 50 pA typical
- Low Input Bias Current ... 200 pA typical

## JORN DUAL-IN-LINE PACKAGE (TOP VIEW)



## description

The TL094 is a quadruple operational amplifier similar in performance to the MC3403 family but with much higher input impedance derived from a FET Input Stage. The N-channel-JFET input stage allows a common-mode input voltage range that includes the negative supply voltage and offers a typical input impedance of  $10^{12}$  ohms, a typical input offset current of 50 picoamperes, and a typical input bias current of 200 picoamperes. The TL094 is designed to operate from a single supply over a range of 3 to 36 volts. Operation from split supplies is also possible provided the difference between the two supplies is 3 to 36 volts. Output voltage range is from  $V_{CC-}$  to 1.5 volts less than  $V_{CC+}$ .

The TL094M is characterized for operation over the full military temperature range of  $-55^\circ\text{C}$  to  $125^\circ\text{C}$ . The TL094I is characterized for operation from  $-40^\circ\text{C}$  to  $85^\circ\text{C}$ . The TL094C is characterized for operation from  $0^\circ\text{C}$  to  $70^\circ\text{C}$ .

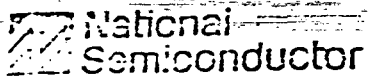
absolute maximum ratings over operating free-air temperature range (unless otherwise noted)

	TL094M	TL094I	TL094C	UNIT
Supply voltage $V_{CC+}$ (see Note 1)	18	18	18	V
Supply voltage $V_{CC-}$ (see Note 1)	-18	-18	-18	V
Supply voltage $V_{CC+}$ with respect to $V_{CC-}$	36	36	36	V
Differential input voltage (see Note 2)	±36	±36	±36	V
Input voltage (see Notes 1 and 3)	-18	-18	18	V
Continuous total dissipation at (or below) $25^\circ\text{C}$ free-air temperature (see Note 4)	J Package 1375 N Package 1150	1025 1150	1025 1150	mW
Operating free-air temperature range	-55 to 125	-40 to 85	0 to 70	$^\circ\text{C}$
Storage temperature range	-65 to 150	-65 to 150	-65 to 150	$^\circ\text{C}$
Lead temperature 1/16 inch (1.6 mm) from case for 60 seconds	300	300	300	$^\circ\text{C}$
Lead temperature 1/18 inch (1.6 mm) from case for 10 seconds	260	260	260	$^\circ\text{C}$

- NOTES: 1. These voltage values are with respect to the midpoint between  $V_{CC+}$  and  $V_{CC-}$ .  
2. Differential voltages are at the noninverting input terminal with respect to the inverting input terminal.  
3. Neither input must ever be more positive than  $V_{CC+}$  or more negative than  $V_{CC-}$  minus 0.3 V.  
4. For operation above  $25^\circ\text{C}$  free-air temperature refer to Dissipation Derating Table. In the J package, TL094M chips are alloy-mounted, TL094I and TL094C chips are glass-mounted.

## DISSIPATION DERATING TABLE

PACKAGE	POWER RATING	DERATING FACTOR	ABOVE $T_A$
J (Alloy-Mounted Chip)	1375 mW	11.0 mW/ $^\circ\text{C}$	$25^\circ\text{C}$
J (Glass-Mounted Chip)	1025 mW	8.2 mW/ $^\circ\text{C}$	$25^\circ\text{C}$
N	1150 mW	9.2 mW/ $^\circ\text{C}$	$25^\circ\text{C}$



**CD4001BM/CD4001BC Quad 2-Input NOR, Buffered B Series Gate**  
**CD4011BM/CD4011BC Quad 2-Input NAND Buffered B Series Gate**

**General Description**

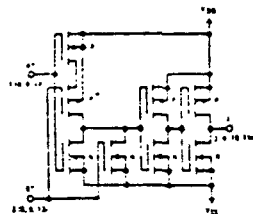
These quad gates are monolithic complementary MOS (CMOS) integrated circuits constructed with N- and P-channel enhancement mode transistors. They have equal source and sink current capabilities and conform to standard B series output drive. The devices also have buffered outputs which improve transfer characteristics by providing very high gain.

All inputs are protected against static discharge with diodes to  $V_{DD}$  and  $V_{SS}$ .

**Features**

- Low power TTL compatibility fan out of 2 driving 74L or 1 driving 74LS
- 5V—10V—15V parametric ratings
- Symmetrical output characteristics
- Maximum input leakage  $1\mu A$  at 15V over full temperature range

**Schematic and Connection Diagrams**

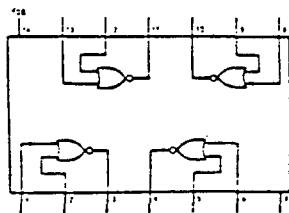


1/4 of device shown

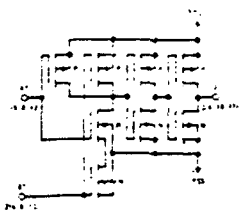
$J = A + B$   
 Logical "1" = High  
 Logical "0" = Low

\*All inputs protected by standard CMOS protection circuit.

CD4001BC/CD4001BM  
 Dual-In-Line and Flat Package



14 pins

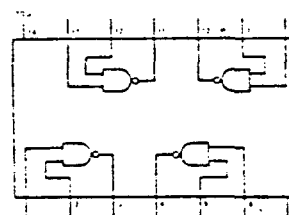


1/4 of device shown

$J = A \cdot B$   
 Logical "1" = High  
 Logical "0" = Low

\*All inputs protected by standard CMOS protection circuit.

CD4011BC/CD4011BM  
 Dual-In-Line and Flat Package



14 pins

**Absolute Maximum Ratings and Operating Conditions**

Operating $V_{DD}$ Range	3 VDC to 15 VDC
Operating Temperature Range	55°C to +125°C
CD4001BM, CD4011BM	40°C to +85°C
CD4001BC, CD4011BC	
Power Dissipation (500 mW)	
Storage Temperature Range	-65°C to +150°C
Temperature (Soldering, 10 seconds)	300°C

**Electrical Characteristics CD4001BM, CD4011BM (Note 2)**

PARAMETER	CONDITIONS	55°C		+25°C			+125°C		UNITS
		MIN	MAX	MIN	TYP	MAX	MIN	MAX	
I <sub>Q</sub> Quiescent Device Current	$V_{DD} = 5V$		0.25		0.004	0.25		7.5	$\mu A$
	$V_{DD} = 10V$		0.50		0.005	0.50		15	$\mu A$
	$V_{DD} = 15V$		1.0		0.006	1.0		30	$\mu A$
V <sub>OL</sub> Low Level Output Voltage	$V_{DD} = 5V$		0.05		0	0.05		0.05	V
	$V_{DD} = 10V$	$ I_O  < 1\mu A$	0.05		0	0.05		0.05	V
	$V_{DD} = 15V$		0.05		0	0.05		0.05	V
V <sub>OH</sub> High Level Output Voltage	$V_{DD} = 5V$	4.95		4.95	5		4.95		V
	$V_{DD} = 10V$	9.95		9.95	10		9.95		V
	$V_{DD} = 15V$	14.95		14.95	15		14.95		V
V <sub>IL</sub> Low Level Input Voltage	$V_{DD} = 5V, V_O = 4.5V$		1.5		2	1.5		1.5	V
	$V_{DD} = 10V, V_O = 9.0V$		3.0		4	3.0		3.0	V
	$V_{DD} = 15V, V_O = 13.5V$		4.0		6	4.0		4.0	V
V <sub>IH</sub> High Level Input Voltage	$V_{DD} = 5V, V_O = 0.5V$		3.5		3.5	3		3.5	V
	$V_{DD} = 10V, V_O = 1.0V$		7.0		7.0	6		7.0	V
	$V_{DD} = 15V, V_O = 1.5V$		11.0		11.0	9		11.0	V
I <sub>OL</sub> Low Level Output Current	$V_{DD} = 5V, V_O = 0.4V$		0.64		0.51	0.88		0.36	mA
	$V_{DD} = 10V, V_O = 0.5V$		1.6		1.3	2.25		0.9	mA
	$V_{DD} = 15V, V_O = 1.5V$		4.2		3.4	8.8		2.4	mA
I <sub>OH</sub> High Level Output Current	$V_{DD} = 5V, V_O = 4.6V$		-0.64		-0.51	0.88		0.36	mA
	$V_{DD} = 10V, V_O = 9.5V$		-1.6		-1.3	2.25		0.9	mA
	$V_{DD} = 15V, V_O = 13.5V$		-4.2		-3.4	8.8		-2.4	mA
Input Current	$V_{DD} = 15V, V_{IN} = 0V$		-0.10		-10 <sup>-5</sup>	-0.10		-1.0	$\mu A$
	$V_{DD} = 15V, V_{IN} = 15V$		0.10		10 <sup>-5</sup>	0.10		1.0	$\mu A$

Note 1: "Absolute Maximum Ratings" are those values beyond which the safety of the device cannot be guaranteed. Except for "Operating Temperature Range" they are not meant to imply that the devices should be operated at these limits. The rules of "Electrical Characteristics" apply for actual device operation.

Note 2: All voltages measured with respect to  $V_{SS}$  unless otherwise specified.

DC Electrical Characteristics CD4001BC, CD4011BC (Note 2)

PARAMETER	CONDITIONS	40°C		+25°C		+85°C		
		MIN	MAX	MIN	TYP	MAX	MIN	MAX
I <sub>DD</sub> Quiescent Device Current	V <sub>DD</sub> = 5V		1		0.004	1		7.5
	V <sub>DD</sub> = 10V		2		0.005	2		15
	V <sub>DD</sub> = 15V		4		0.006	4		30
V <sub>OL</sub> Low Level Output Voltage	V <sub>DD</sub> = 5V		0.05		0	0.05		0.05
	V <sub>DD</sub> = 10V		0.05		0	0.05		0.05
	V <sub>DD</sub> = 15V		0.05		0	0.05		0.05
V <sub>OH</sub> High Level Output Voltage	V <sub>DD</sub> = 5V	4.95		4.95	5		4.95	
	V <sub>DD</sub> = 10V	9.95		9.95	10		9.95	
	V <sub>DD</sub> = 15V	14.95		14.95	15		14.95	
V <sub>IL</sub> Low Level Input Voltage	V <sub>DD</sub> = 5V, V <sub>O</sub> = 4.5V		1.5		2		1.5	
	V <sub>DD</sub> = 10V, V <sub>O</sub> = 9.0V		3.0		4		3.0	
	V <sub>DD</sub> = 15V, V <sub>O</sub> = 13.5V		4.0		6		4.0	
V <sub>IH</sub> High Level Input Voltage	V <sub>DD</sub> = 5V, V <sub>O</sub> = 0.5V	3.5		3.5	3		3.5	
	V <sub>DD</sub> = 10V, V <sub>O</sub> = 1.0V	7.0		7.0	6		7.0	
	V <sub>DD</sub> = 15V, V <sub>O</sub> = 1.5V	11.0		11.0	9		11.0	
I <sub>OL</sub> Low Level Output Current	V <sub>DD</sub> = 5V, V <sub>O</sub> = 0.4V		0.52		0.44	0.88		0.36
	V <sub>DD</sub> = 10V, V <sub>O</sub> = 0.5V		1.3		1.1	2.25		0.9
	V <sub>DD</sub> = 15V, V <sub>O</sub> = 1.5V		3.6		3.0	8.8		2.4
I <sub>OH</sub> High Level Output Current	V <sub>DD</sub> = 5V, V <sub>O</sub> = 4.6V	-0.52		-0.44	-0.88		-0.36	
	V <sub>DD</sub> = 10V, V <sub>O</sub> = 9.5V	-1.3		-1.1	-2.25		-0.9	
	V <sub>DD</sub> = 15V, V <sub>O</sub> = 13.5V	-3.6		-3.0	-8.8		-2.4	
I <sub>IN</sub> Input Current	V <sub>DD</sub> = 15V, V <sub>IN</sub> = 0V		-0.30		-10 <sup>-5</sup>	-0.30		-1.0
	V <sub>DD</sub> = 15V, V <sub>IN</sub> = 15V		0.30		10 <sup>-5</sup>	0.30		1.0

AC Electrical Characteristics CD4001BC, CD4001BM

T<sub>A</sub> = 25°C, Input t<sub>r</sub>: t<sub>f</sub> = 20 ns, C<sub>L</sub> = 50 pF, R<sub>L</sub> = 200k. Typical temperature coefficient is 0.3%/°C.

PARAMETER	CONDITIONS	TYP	MAX	UNITS
t <sub>PHL</sub> Propagation Delay Time, High-to-Low Level	V <sub>DD</sub> = 5V	120	250	ns
	V <sub>DD</sub> = 10V	50	100	ns
	V <sub>DD</sub> = 15V	35	70	ns
t <sub>PLH</sub> Propagation Delay Time, Low-to-High Level	V <sub>DD</sub> = 5V	110	250	ns
	V <sub>DD</sub> = 10V	50	100	ns
	V <sub>DD</sub> = 15V	35	70	ns
t <sub>RHL</sub> , t <sub>LHL</sub> Transition Time	V <sub>DD</sub> = 5V	90	200	ns
	V <sub>DD</sub> = 10V	50	100	ns
	V <sub>DD</sub> = 15V	40	80	ns
C <sub>IN</sub> Average Input Capacitance	Any Input	5	7.5	pF
C <sub>PD</sub> Power Dissipation Capacity	Any Gate	14		pF

Electrical Characteristics CD4011BC, CD4011BM

T<sub>A</sub> = 25°C, Input t<sub>r</sub>: t<sub>f</sub> = 20 ns, C<sub>L</sub> = 50 pF, R<sub>L</sub> = 200k. Typical Temperature Coefficient is 0.3%/°C.

PARAMETER	CONDITIONS	TYP	MAX	UNITS
t <sub>PHL</sub> Propagation Delay, High-to-Low Level	V <sub>DD</sub> = 5V	120	250	ns
	V <sub>DD</sub> = 10V	50	100	ns
	V <sub>DD</sub> = 15V	35	70	ns
t <sub>PLH</sub> Propagation Delay, Low-to-High Level	V <sub>DD</sub> = 5V	85	250	ns
	V <sub>DD</sub> = 10V	40	100	ns
	V <sub>DD</sub> = 15V	30	70	ns
t <sub>RHL</sub> , t <sub>LHL</sub> Transition Time	V <sub>DD</sub> = 5V	90	200	ns
	V <sub>DD</sub> = 10V	50	100	ns
	V <sub>DD</sub> = 15V	40	80	ns
C <sub>IN</sub> Average Input Capacitance	Any Input	5	7.5	pF
C <sub>PD</sub> Power Dissipation Capacity	Any Gate	14		pF

Typical Performance Characteristics

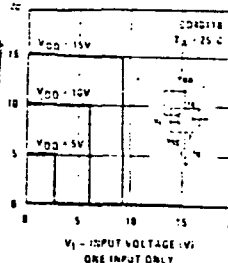


FIGURE 1. Typical Transfer Characteristics

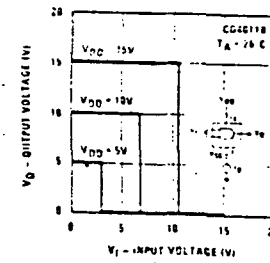


FIGURE 2. Typical Transfer Characteristics

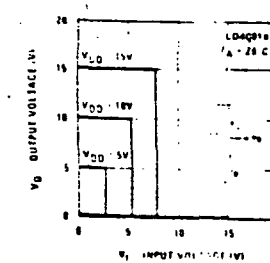


FIGURE 3. Typical Transfer Characteristics

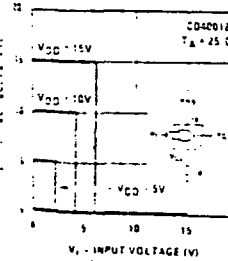


FIGURE 4. Typical Transfer Characteristics

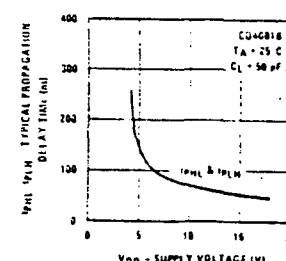


FIGURE 5

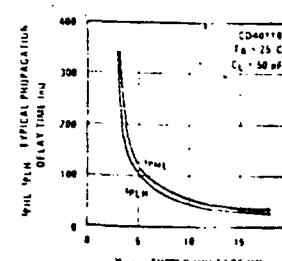


FIGURE 6



# CD40152M/CD40153C DUAL 4-BIT Static Shift Register

## General Description

The CD40152M/CD40153C contains two identical, 4-stage, serial-input/parallel-output registers with independent "Data," "Clock," and "Reset" inputs. The logic level present at the input of each stage is transferred to the output of that stage at each positive-going clock transition. A logic high on the "Reset" input resets all four stages covered by that input. All inputs are protected from static discharge by a series resistor and diode clamps to V<sub>DD</sub> and V<sub>SS</sub>.

- Low power TTL compatibility fan out of 2 driving 74L or 1 driving 74LS
- Medium speed operation 8 MHz (typ) clock rate
- Fully static design @ V<sub>DD</sub> - V<sub>SS</sub> = 10 Volts

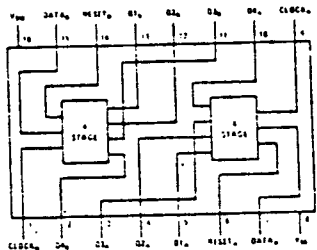
## Applications

- Serial-input/parallel-output data queuing
- Serial to parallel data conversion
- General purpose register

## Features

- Wide supply voltage range 3.0V to 18V
- High noise immunity 0.45 V<sub>DD</sub> (typ.)

## Connection Diagram and Truth Table

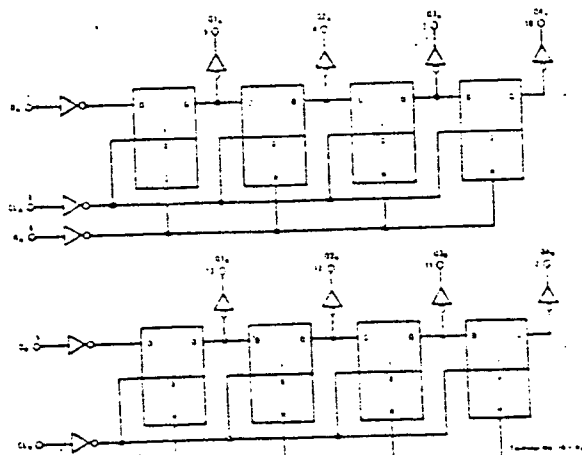


CL	D	R	Q <sub>1</sub>	Q <sub>2</sub>	Q <sub>3</sub>	Q <sub>4</sub>
0	0	0	0	0	0	0
1	0	0	1	0	0	0
X	0	0	0	0	0	0
X	X	1	0	0	0	0

(No change)

0 Low/no change  
X Don't care case

## Logic Diagrams



## Maximum Ratings (Notes 1 and 2)

Supply Voltage	-0.5 to +18 V <sub>DC</sub>
Input Voltage	-0.5 to V <sub>DD</sub> + 0.5 V <sub>DC</sub>
Operating Temperature Range	-65 to +150 °C
Power Dissipation	500 mW
Soldering Temperature (Soldering, 10 seconds)	300 °C

## Recommended Operating Conditions

DC Supply Voltage	+3 to +15 V <sub>DC</sub>
Input Voltage	0 to V <sub>DD</sub> V <sub>DC</sub>
Operating Temperature Range	
CD40152M	-55 °C to +125 °C
CD40153C	-40 °C to +85 °C

## DC Electrical Characteristics (Note 2) — CD40152M

Parameter	Conditions	-55 °C		25 °C		125 °C		Units	
		Min	Max	Min	Typ	Max	Min		Max
I <sub>DD</sub> Quiescent Device Current	V <sub>DD</sub> = 5V		5		0.005		5	150	μA
	V <sub>DD</sub> = 10V		10		0.010		10	300	μA
	V <sub>DD</sub> = 15V		20		0.015		20	600	μA
V <sub>OL</sub> Low Level Output Voltage	V <sub>DD</sub> = 5V		0.05		0		0.05	0.05	V
	V <sub>DD</sub> = 10V		0.05		0		0.05	0.05	V
	V <sub>DD</sub> = 15V		0.05		0		0.05	0.05	V
V <sub>OH</sub> High Level Output Voltage	V <sub>DD</sub> = 5V	4.95		4.95	5		4.95		V
	V <sub>DD</sub> = 10V	9.95		9.95	10		9.95		V
	V <sub>DD</sub> = 15V	14.95		14.95	15		14.95		V
V <sub>IH</sub> High Level Input Voltage	V <sub>DD</sub> = 5V, V <sub>O</sub> = 0.5V or 4.5V		1.5		2.25		1.5	1.5	V
	V <sub>DD</sub> = 10V, V <sub>O</sub> = 1.0V or 9.0V		3.0		4.50		3.0	3.0	V
	V <sub>DD</sub> = 15V, V <sub>O</sub> = 1.5V or 13.5V		4.0		6.75		4.0	4.0	V
V <sub>IL</sub> Low Level Input Voltage	V <sub>DD</sub> = 5V, V <sub>O</sub> = 0.5V or 4.5V	0.5		3.5	2.75		3.5		V
	V <sub>DD</sub> = 10V, V <sub>O</sub> = 1.0V or 9.0V	7.0		7.0	5.50		7.0		V
	V <sub>DD</sub> = 15V, V <sub>O</sub> = 1.5V or 13.5V	11.0		11.0	8.25		11.0		V
I <sub>OL</sub> Low Level Output Current	V <sub>DD</sub> = 5V, V <sub>O</sub> = 0.4V	0.64		0.51	0.65		0.36		mA
	V <sub>DD</sub> = 10V, V <sub>O</sub> = 0.5V	1.6		1.3	2.25		0.9		mA
	V <sub>DD</sub> = 15V, V <sub>O</sub> = 1.5V	4.2		3.4	6.8		2.4		mA
I <sub>OH</sub> High Level Output Current	V <sub>DD</sub> = 5V, V <sub>O</sub> = 4.6V	-0.64		-0.51	-0.88		-0.36		mA
	V <sub>DD</sub> = 10V, V <sub>O</sub> = 9.5V	-1.6		-1.3	-2.25		-0.9		mA
	V <sub>DD</sub> = 15V, V <sub>O</sub> = 13.5V	-4.2		-3.4	-6.8		-2.4		mA
I <sub>IN</sub> Input Current	V <sub>DD</sub> = 15V, V <sub>IN</sub> = 0V		-0.1		-10 <sup>-5</sup>		-0.1	-1.0	μA
	V <sub>DD</sub> = 15V, V <sub>IN</sub> = 15V		0.1		10 <sup>-5</sup>		0.1	1.0	μA

CD40152M/CD40153C

DC Electrical Characteristics (Note 2) - CD4015BC

Parameter	Conditions	-40°C		25°C		85°C		Units
		Min	Max	Min	Typ	Max	Min	
I <sub>DD</sub> Quiescent Device Current	V <sub>DD</sub> = 5V		20		0.005	20		150
	V <sub>DD</sub> = 10V		40		0.010	40		300
	V <sub>DD</sub> = 15V		80		0.015	20		600
V <sub>OL</sub> Low Level Output Voltage	V <sub>DD</sub> = 5V		0.05		0	0.05		0.05
	V <sub>DD</sub> = 10V		0.05		0	0.05		0.05
	V <sub>DD</sub> = 15V		0.05		0	0.05		0.05
V <sub>OH</sub> High Level Output Voltage	V <sub>DD</sub> = 5V	4.95		4.95	5		4.95	
	V <sub>DD</sub> = 10V	9.95		9.95	10		9.95	
	V <sub>DD</sub> = 15V	14.95		14.95	15		14.95	
V <sub>IL</sub> Low Level Input Voltage	V <sub>DD</sub> = 5V, V <sub>O</sub> = 0.5V or 4.5V		1.5		2.25	1.5		1.5
	V <sub>DD</sub> = 10V, V <sub>O</sub> = 1.0V or 9.0V		3.0		4.50	3.0		3.0
	V <sub>DD</sub> = 15V, V <sub>O</sub> = 1.5V or 13.5V		4.0		6.75	4.0		4.0
V <sub>IH</sub> High Level Input Voltage	V <sub>DD</sub> = 5V, V <sub>O</sub> = 0.5V; or 4.5V	3.5		3.5	2.75		3.5	
	V <sub>DD</sub> = 10V, V <sub>O</sub> = 1.0V or 9.0V	7.0		7.0	5.50		7.0	
	V <sub>DD</sub> = 15V, V <sub>O</sub> = 1.5V or 13.5V	11.0		11.0	8.25		11.0	
I <sub>OL</sub> Low Level Output Current	V <sub>DD</sub> = 5V, V <sub>O</sub> = 0.4V	0.52		0.44	0.88		0.36	
	V <sub>DD</sub> = 10V, V <sub>O</sub> = 0.5V	1.3		1.1	2.25		0.9	
	V <sub>DD</sub> = 15V, V <sub>O</sub> = 1.5V	3.6		3.0	8.8		2.4	
I <sub>OH</sub> High Level Output Current	V <sub>DD</sub> = 5V, V <sub>O</sub> = 4.6V	-0.52		-0.44	-0.88		-0.36	
	V <sub>DD</sub> = 10V, V <sub>O</sub> = 9.5V	-1.3		-1.1	-2.25		-0.9	
	V <sub>DD</sub> = 15V, V <sub>O</sub> = 13.5V	-3.6		-3.0	-8.8		-2.4	
I <sub>IN</sub> Input Current	V <sub>DD</sub> = 15V, V <sub>IN</sub> = 0V		-0.3		-10 <sup>-5</sup>	-0.3		-1.0
	V <sub>DD</sub> = 15V, V <sub>IN</sub> = 15V		0.3		10 <sup>-5</sup>	0.3		1.0

Note 1: "Absolute Maximum Ratings" are those values beyond which the safety of the device cannot be guaranteed, they are not meant to imply that the devices should be operated at these limits. The tables of "Recommended Operating Conditions" and "Electrical Characteristics" provide conditions for actual device operation.

Note 2: V<sub>SS</sub> = CV unless otherwise specified.

AC Electrical Characteristics (Note 2) - CD4015BC  
T<sub>A</sub> = 25°C, C<sub>L</sub> = 50 pF, R<sub>L</sub> = 200 kΩ, t<sub>PLH</sub> = t<sub>PHL</sub> = 20 ns, unless otherwise specified.

Clocked Operation	Parameter	Conditions	Min	Typ	Max	Units
I <sub>PHL</sub> , I <sub>PLH</sub>	Propagation Delay Time	V <sub>DD</sub> = 5V		230	350	ns
		V <sub>DD</sub> = 10V		80	160	ns
		V <sub>DD</sub> = 15V		60	120	ns
t <sub>RHL</sub> , t <sub>RHL</sub>	Transition Time	V <sub>DD</sub> = 5V		100	200	ns
		V <sub>DD</sub> = 10V		50	100	ns
		V <sub>DD</sub> = 15V		40	40	ns
t <sub>WL</sub> , t <sub>WM</sub>	Minimum Clock Pulse-Width	V <sub>DD</sub> = 5V		160	250	ns
		V <sub>DD</sub> = 10V		60	110	ns
		V <sub>DD</sub> = 15V		50	100	ns
t <sub>rCL</sub> , t <sub>rCL</sub>	Clock Rise and Fall Time	V <sub>DD</sub> = 5V			15	μs
		V <sub>DD</sub> = 10V			15	μs
		V <sub>DD</sub> = 15V			15	μs
t <sub>SU</sub>	Minimum Data Set-Up Time	V <sub>DD</sub> = 5V		50	100	ns
		V <sub>DD</sub> = 10V		20	40	ns
		V <sub>DD</sub> = 15V		15	30	ns
f <sub>CL</sub>	Maximum Clock Frequency	V <sub>DD</sub> = 5V	2	3.5		MHz
		V <sub>DD</sub> = 10V	4.5	8		MHz
		V <sub>DD</sub> = 15V	6	11		MHz
C <sub>IN</sub>	Input Capacitance	Clock Input	7.5		10	pF
		Other Inputs	5		7.5	pF
<b>Reset Operation</b>						
t <sub>PHL(R)</sub>	Propagation Delay Time	V <sub>DD</sub> = 5V		200	400	ns
		V <sub>DD</sub> = 10V		100	200	ns
		V <sub>DD</sub> = 15V		60	160	ns
t <sub>WH(R)</sub>	Minimum Reset Pulse Width	V <sub>DD</sub> = 5V		135	250	ns
		V <sub>DD</sub> = 10V		40	80	ns
		V <sub>DD</sub> = 15V		30	60	ns

operaciones que transfieren los bits uno por uno en una sola línea. Naturalmente, las operaciones de entrada/salida en serie son mucho más lentas pero a su vez requieren una cantidad menor de componentes que las operaciones en paralelo.

### Interrupciones

Muchos procesadores centrales incluyen la posibilidad de interrupción como una manera de mejorar la eficiencia de procesamiento. Considere el caso de un computador que está procesando un gran volumen de datos que va a ser transferido a la impresora por porciones. El CPU puede enviar un byte de datos en un ciclo de máquina pero la impresora puede que requiera un tiempo equivalente a muchos ciclos de máquina para realizar la impresión del carácter especificado por el byte de datos. El CPU podría, mientras tanto, permanecer inactivo esperando a que la impresora acepte el próximo byte de datos. Si el computador incluye la función de interrupción, el CPU puede continuar el procesamiento después de enviar el byte de datos a la impresora. Cuando la impresora esté lista para aceptar el siguiente byte de datos, puede enviar una solicitud de interrupción. Cuando el CPU acepta la interrupción, suspende la ejecución del programa principal y automáticamente salta a una rutina que envía a la impresora el siguiente byte de datos. Después del envío de este byte, el CPU continúa ejecutando el programa principal. Nóte que esto, en principio, es muy similar al salto resultante de llamar (call) una rutina a excepción de que el salto se origina externamente y no en el programa.

Es posible tener estructuras de interrupción más complejas en las cuales varios dispositivos pueden interrumpir al mismo procesador pero con diferentes niveles de prioridad. El procesamiento interruptivo es una importante característica que permite la máxima utilización de la capacidad del procesador para un alto rendimiento del sistema.

### Detención de la Operación del Procesador (Hold)

Otra característica importante que aumenta el rendimiento del procesador es poder detenerlo (Hold). Esto permite las operaciones de acceso directo a memoria (Direct Memory Access — DMA).

En las operaciones ordinarias de entrada y salida el procesador supervisa todas las transferencias de datos. La información con destino a la memoria se transfiere del dispositivo de entrada al procesador y después del procesador a la localización seleccionada en la memoria. De manera

similar, la información con destino a los dispositivos de salida se transfiere a través del procesador.

Sin embargo, algunos dispositivos periféricos son capaces de transferir información que entre o salga de la memoria con mayor rapidez de la que puede realizar el procesador. Si se utiliza este tipo de dispositivo para transferir una cantidad apreciable de datos, el rendimiento del sistema es aumentado al permitir que el dispositivo realice directamente las transferencias. El procesador tiene que suspender temporalmente su operación durante estas transferencias directas para evitar los conflictos que pudieran presentarse si el procesador y un dispositivo periférico intentan tener acceso simultáneamente a la memoria. Por esta razón, es necesario que algunos procesadores incluyan esta característica de poder suspender su operación.

## CPU — MICROPROCESADOR

Las unidades NCR basadas en el empleo de microprocesadores utilizan la pastilla Intel 8080 como la unidad central de procesamiento.

El 8080 es una unidad central de procesamiento (CPU) de 8 bits en paralelo utilizada en los sistemas de computadores digitales de aplicación general. Está fabricado en una pastilla de integración a gran escala (LSI), ver Figura 2, utilizándose el proceso Metal-Oxido-Semiconductor de canal n con compuerta de silicio (n-channel silicon gate MOS) de Intel. El 8080 transfiere datos e información del estado interno por medio de un Canal de Datos de 8 bits (D0-D7) bidireccional (transferencias en ambos sentidos) de tres estados (bidireccional 3-state Data Bus). Para transmitir las direcciones de memoria y de selección de periféricos utiliza un Canal de Direcciones de 16 bits (A0-A15) de tres estados (3-state Address Bus). El 8080 tiene seis salidas para señales de sincronismo y control (SYNC, DBIN, WAIT, WR, HLDA e INTE), cuatro entradas de control (READY, HOLD, INT y RESET), cuatro entradas de potencia (+12V, +5V, -5V y GRD) y dos entradas para los pulsos de sincronización (01 y 02).

### ARQUITECTURA DEL CPU 8080

El CPU 8080 consta de las siguientes unidades operativas:

- Juego de Registros (Register Array) y Lógica de Dirección

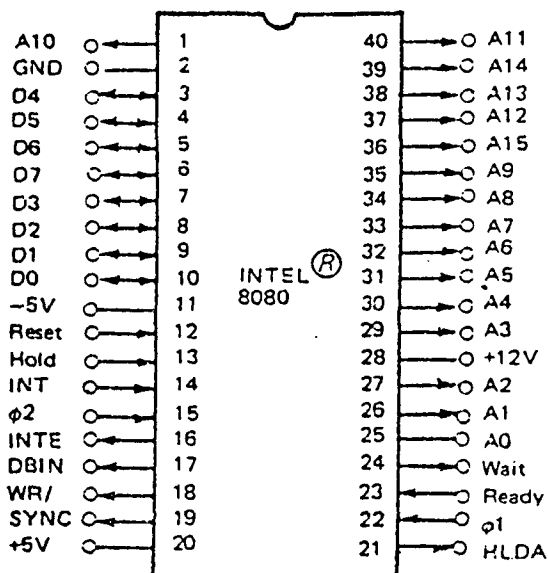


Figura 2 Denominación de Terminales del 8080

Los seis registros de utilidad general se pueden usar como registros sencillos (8 bits) o en pares (16 bits). El par de registros W-Z de almacenamiento temporal no se puede seleccionar por dirección y se utiliza solamente para la ejecución interna de instrucciones.

Los bytes de datos (8-bits) se pueden transferir entre el canal interno y el juego de registros por medio de un multiplexor de selección de registros (Register-Select Multiplexer). Pueden realizarse transferencias de 16 bits entre el juego de registros y el Retenedor de Direcciones (Address Latch) o el Circuito de Incremento/Decremento (Incrementer/Decrementer Circuit). El retenedor de direcciones recibe la información contenida en uno cualquiera de los 3 pares de registros y activa los 16 Circuitos de Salida de Direcciones A0-A15 (Address Output Buffers) y el circuito de incremento/decremento recibe la información del retenedor de dirección y la envía a los registros. El valor de la información en los 16 bits puede ser aumentado, disminuido o sencillamente transferido entre registros.

### Unidad Aritmética/Lógica (ALU)

La unidad aritmética/lógica contiene los siguientes registros:

- Un Acumulador (Accumulator) de 8 bits
- Un Acumulador Temporal (Temporary Accumulator-ALT) de 8 bits
- Un Registro de Banderas de 5 bits: cero (Zero), Transferencia (Carry), Signo (Sign), Paridad (Parity) y Transferencia Auxiliar (Auxiliary Carry)
- Un Registro Temporal (Temporary Register — TMP) de 8 bits

Las operaciones aritméticas y de rotación de información se realizan en el ALU. Este recibe información del registro temporal TMP, del acumulador temporal ACT y de la bandera de transferencia (carry). Los resultados de la operación se pueden transferir al canal interno o al acumulador; el ALU también puede afectar el registro de banderas.

El registro temporal TMP recibe información del canal interno y puede enviar toda o parcialmente la información al ALU, al registro de banderas o al canal interno.

El acumulador ACC puede cargarse con datos del canal interno o del ALU y puede transferir datos al acumulador temporal ACT y al canal interno. El

- Unidad Aritmética y Lógica (ALU)
- Registro de Instrucción (Instruction Register) y Sección de Control
- Circuito Amortiguador de Datos (Data Bus Buffer) bi-direccional de tres estados

La Figura 3 ilustra los bloques operativos en el CPU 8080.

### Registros

El conjunto de 6 registros de 16 bits está organizado en una sección de memoria estática de acceso aleatorio (Random Access Memory-RAM):

- Contador del Programa (Program Counter — PC)
- Registro Stack (Stack Pointer — SP)
- Seis Registros de utilidad general organizados en pares llamados B-C, D-E y H-L
- Un par de registros temporales llamados W-Z

El contador del programa mantiene la dirección de la localización en la memoria externa que contiene la instrucción corriente del programa y automáticamente es incrementado durante la ejecución de cada instrucción. El registro stack, o SP, mantiene la dirección de la siguiente localización disponible en la memoria Stack. Este registro se puede inicializar para que especifique cualquier área de la memoria de escritura/lectura como la sección de memoria Stack. El registro SP disminuye la dirección cuando se ponen datos en la memoria stack (push) (la pila de datos aumenta con disminución de dirección) y aumenta la dirección cuando se sacan.



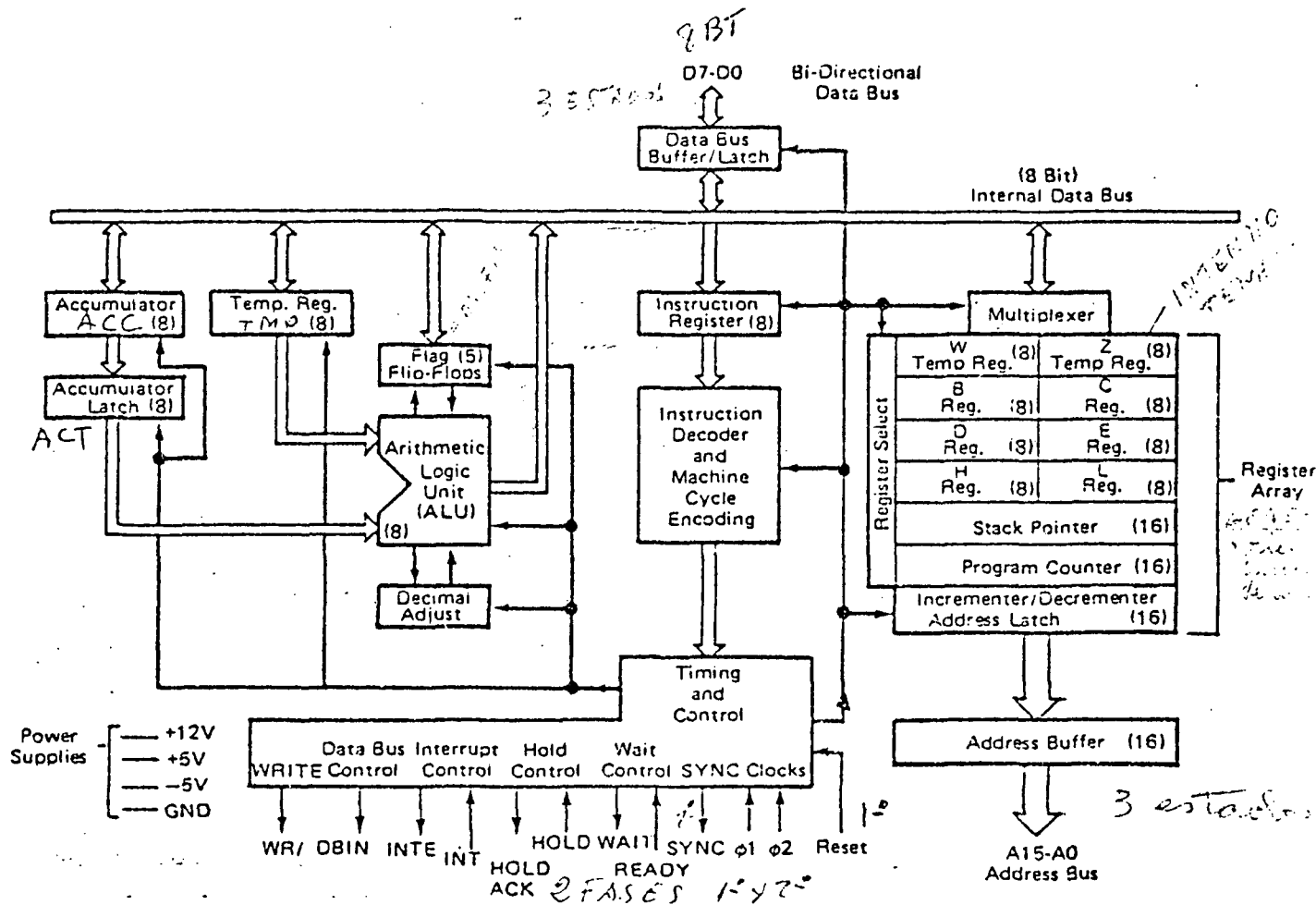


Figura 3 CPU 8080 - Diagrama por Bloques Operativos

contenido del acumulador ACC y de la transferencia auxiliar pueden verificarse y determinar la corrección decimal durante la ejecución de la instrucción DAA (Decimal Adjust Accumulator).

### Registro de Instrucción y Control

Durante la obtención de una instrucción (Instruction Fetch), el primer byte de la instrucción (el cual contiene el código de operación — OP) se transfiere del canal interno al registro de instrucción.

El contenido del registro de instrucción a su vez, queda disponible para el Decodificador de Instrucción (Instruction Decoder). Las salidas del decodificador, en combinación con varias señales de sincronismo, proporcionan las señales de control para el juego de registros, el ALU y los circuitos amortiguadores de datos. También, las salidas del decodificador y las señales externas de control entran en la sección de sincronismo y control de estado (Timing and State Control) la cual genera los estados y señales de sincronismo del ciclo.

### Circuito de Salida/Entrada del Canal de Datos

Estos circuitos componen una sección (Data Bus Buffer) de 8 bits, bidireccional, de tres estados que se utiliza para separar el canal interno del CPU del canal de datos externo (D0-D7). En operaciones de salida, el contenido del canal interno se carga en un retenedor de 8 bits el cual envía la información a los circuitos de salida del canal de datos. Los circuitos de salida se inhabilitan durante las operaciones de entrada y aquellas que no transfieran información.

En operaciones de entrada, la información contenida en el canal externo de datos se transfiere al canal interno. El canal interno se precarga al comienzo de cada estado interno exceptuando el estado de transferencia T3 (posteriormente descrito).

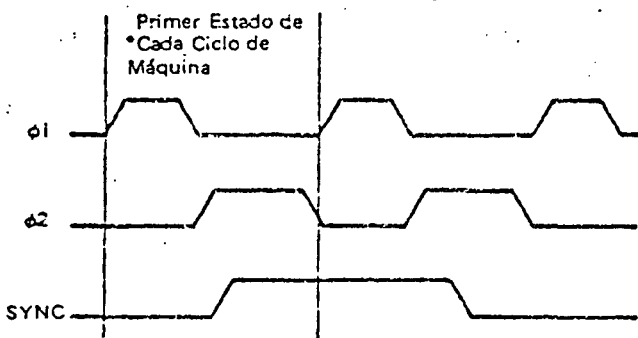
### CICLO DEL PROCESADOR

Un ciclo de instrucción se define como el tiempo requerido para obtener (fetch) y ejecutar una instrucción. Durante la obtención, la instrucción seleccionada, de uno, dos o tres bytes de longitud, es leída de la memoria y cargada en el CPU.

Durante la fase de ejecución, la instrucción se decodifica y determina acciones específicas de procesamiento.

Cada ciclo de instrucción consiste de uno, dos, tres, cuatro o cinco ciclos de máquina. Se requiere un ciclo de máquina cada vez que el CPU tiene acceso a la memoria o a un puerto. La porción del ciclo para obtención de la instrucción requiere un ciclo de máquina por cada byte de la instrucción. La duración de la porción de ejecución depende de la instrucción en ejecución. Algunas instrucciones solamente requieren los ciclo de máquina necesarios para la obtención; otras instrucciones, en cambio, requieren ciclos de máquina adicionales para escribir o leer la memoria y para enviar o recibir información de los dispositivos de entrada/salida. La instrucción DAD (Double Add) es una excepción ya que requiere dos ciclos de máquina adicionales para completar la suma de un par interno de registros.

Cada ciclo de máquina consiste de tres, cuatro o cinco estados. Un estado es la unidad mínima de actividad de procesamiento y se define como el intervalo de tiempo entre dos transiciones positivas del pulso de sincronización (Clock Pulse)  $\phi 1$ . El 8080 es regulado por dos pulsos de fase suministrados por un circuito oscilador externo que genera dos pulsos de sincronización no simultáneos llamados  $\phi 1$  y  $\phi 2$ . El pulso  $\phi 1$  divide los ciclos de máquina en estados. La lógica de sincronismo dentro del 8080 utiliza los pulsos de sincronización para generar el pulso SYNC, el cual identifica el comienzo de cada ciclo de máquina. El pulso SYNC se genera cuando ocurre la transición positiva del pulso  $\phi 2$  como muestra la Figura 4.



\*SYNC no ocurre durante los ciclos de máquina segundo y tercero de la instrucción DAD debido a que son utilizados para la suma de uno de los pares de registros internos

Figura 4 Tiempo de las Señales  $\phi 1$ ,  $\phi 2$  y SYNC

La duración de los estados que se ha definido anteriormente tiene tres excepciones. Estas son las

duraciones del estado de espera WAIT, del estado de suspensión HLDA (Hold) y del estado de detención HLTA (Halt) los cuales se describen en secciones posteriores. Debido a que los estados WAIT, HLDA y HLTA dependen de condiciones externas pueden tener duraciones indeterminadas. Estos estados tienen que estar sincronizados con los pulsos de sincronización y por consiguiente el periodo de su duración es un múltiplo del periodo de los pulsos de sincronización.

En resumen: cada periodo del pulso de sincronización determina un estado; de 3 a 5 estados componen un ciclo de máquina y de 1 a 5 ciclos de máquina componen un ciclo de instrucción. Un ciclo completo de instrucción requiere de 4 a 18 estados dependiendo de la clase de instrucción.

### Identificación del Ciclo de Máquina

Con excepción de la instrucción DAD, un solo factor determina cuantos ciclos de máquina son necesarios para la ejecución de un ciclo de instrucción: el número de veces que el procesador tiene que seleccionar la memoria o un dispositivo periférico para obtener y ejecutar la instrucción. Al igual que muchos otros procesadores, la construcción del 8080 permite la transferencia de solamente una dirección en cada ciclo de máquina. De esta manera, si la obtención y ejecución de una instrucción requiere dos accesos a memoria, el ciclo de esta instrucción se compone de dos ciclos de máquina; si la instrucción requiere 5 accesos a memoria, entonces el ciclo de instrucción contiene 5 ciclos de máquina.

Cada ciclo de instrucción incluye por lo menos un acceso a memoria durante el cual se obtiene la instrucción. Un ciclo de instrucción debe incluir la obtención de la instrucción aún cuando en su ejecución no sea necesario tener más accesos a la memoria. Por consiguiente, el primer ciclo de máquina en cada ciclo de instrucción es un ciclo de obtención (FETCH MACHINE CYCLE) después del cual no aplican reglas fijas ya que los ciclos subsiguientes dependen de la instrucción obtenida.

Refiérase a la Tabla 4 y considere algunos ejemplos. La instrucción de suma de registros Add r (Add-Register) es una instrucción que requiere un solo ciclo de máquina (FETCH-M1) para completarse. Con esta instrucción de un byte, el contenido de uno de los seis pares de registros de utilidad general del CPU es sumado al contenido existente en el acumulador. Puesto que toda la información necesaria para la ejecución de la instrucción está contenida en los 8 bits del código de operación se necesita un solo acceso a la memoria.

Se requieren 3 estados para leer la instrucción en la memoria y uno adicional para realizar la función de suma de la instrucción. El ciclo de instrucción completo, en consecuencia, requiere un solo ciclo de máquina compuesto de cuatro estados, o sean cuatro períodos de los pulsos de sincronización externos. Suponga que ahora, desea sumar al acumulador el contenido de una localización específica de memoria utilizando la instrucción ADD M (Add Memory). Aunque es similar a la instrucción del ejemplo anterior, esta instrucción usa varios pasos adicionales. Es necesario un ciclo de máquina suplementario para seleccionar la localización deseada en la memoria.

La siguiente es la secuencia de los eventos de la instrucción. El procesador obtiene primero esta instrucción de un byte seleccionado una localización de la memoria por medio de la dirección contenida en el contador del programa PC. Esto requiere 3 estados. El byte (8 bits) de la instrucción obtenida durante el ciclo (FETCH) se carga en el registro de instrucción del CPU y se utiliza para determinar las actividades durante el resto del ciclo de instrucción. Después, el procesador envía a la memoria la dirección contenida en el par de registros H y L. El byte enviado por la memoria durante el ciclo de lectura (MEMORY READ CYCLE) se carga en un registro temporal en el CPU 8080. Cuando esto se realiza han transcurrido ya 3 estados o períodos de sincronismo adicionales. Durante el séptimo y último estado, se suma el contenido del registro temporal al contenido del acumulador. Así, el ciclo de la instrucción "ADD M" se ha ejecutado en 2 ciclos de máquina consistentes de un total de 7 estados.

Como ejemplo del caso de extremo opuesto, se tiene la instrucción SHLD (Store H & L Direct) la cual requiere 5 ciclos de máquina. Durante la ejecución de esta instrucción, el contenido del par de registros H y L del procesador se almacena en dos localizaciones adyacentes de memoria. Las localizaciones están indicadas por los dos bytes de instrucción que siguen inmediatamente después del código de operación. La secuencia de los eventos que ocurren al ejecutarse la instrucción es:

- Ciclo de máquina para la obtención de la instrucción (FETCH Machine Cycle) compuesto de cuatro estados. Durante los primeros tres estados del ciclo, el procesador obtiene la instrucción en la localización seleccionada por el contador del programa. Después incrementa la dirección contenida en el contador del programa. El cuarto estado se utiliza para la decodificación interna de la instrucción.

- Ciclo de máquina para lectura de memoria (MEMORY READ Machine Cycle) compuesto de tres estados. Durante este ciclo, el byte cuya localización específica la dirección en el contador del programa es leído de la memoria y almacenado en el registro Z del procesador. Se incrementa nuevamente el contador de programa.
- Otro ciclo de máquina para lectura de memoria (MEMORY READ Machine Cycle) compuesto de tres estados durante el cual se lee en la memoria el byte cuya localización específica la dirección en el contador del programa y después se almacena en el registro W del procesador. El contador del programa se incrementa preparándose para la obtención de la próxima instrucción.
- Ciclo de máquina para escritura en memoria (MEMORY WRITE Machine Cycle) compuesto de tres estados, durante el cual se transfiere el contenido del registro L a la localización en memoria cuya dirección específica el par de registros W y Z. Un estado subsiguiente a los estados de transferencia se utiliza para incrementar el par de registros W y Z de modo que indique la localización siguiente que va a recibir datos en la memoria.
- Ciclo de máquina para escritura en memoria (MEMORY WRITE Machine Cycle) compuesto de tres estados durante el cual se transfiere el contenido del registro H a la localización en memoria cuya dirección específica el par de registros W y Z.

En resumen, el ciclo de la instrucción "SHLD" se compone de 5 ciclos de máquina y requiere 16 estados para su ejecución.

La mayoría de los ciclos de instrucción se compone de un número de ciclos de máquina que está comprendido dentro de los dos extremos ilustrados por las instrucciones "ADD r" y "SHLD". Instrucciones tales como INP (Input) y OUT (Output) para entrada y salida de datos respectivamente, por ejemplo, requieren tres ciclos de máquina: Obtención de la Instrucción (FETCH); Lectura de Memoria (MEMORY READ) para conseguir la dirección que selecciona el periférico; por último, Entrada (INPUT) o Salida (OUTPUT) para transferir los datos completando así la instrucción.

Aunque ninguna instrucción incluye más de 5 ciclos de máquina, los siguientes diez ciclos de máquina diferentes pueden formar parte de un ciclo de instrucción.

ESTADOS

- Obtención de la Instrucción (FETCH)
- Lectura de Memoria (MEMORY READ)
- Escritura en Memoria (MEMORY WRITE)
- Lectura del Area STACK en Memoria (STACK READ)
- Escritura en el Area STACK en Memoria (STACK WRITE)
- Transferencia de Entrada (INPUT)
- Transferencia de Salida (OUTPUT)
- Interrupción (INTERRUPT)
- Detención (HALT)
- Detención-Interrupción (HALT-INTERRUPT)

Los ciclos de máquina que ocurren durante la ejecución de un determinado ciclo de instrucción dependen del tipo de instrucción exceptuando el primer ciclo de máquina que es siempre el de obtención de la instrucción (FETCH)

El procesador identifica el ciclo de máquina en progreso emitiendo una palabra de estado (Status Word) de 8 bits durante el primer estado (state) de cada ciclo de máquina. La información actual del estado se envía por medio de las líneas de datos D0-D7 del 8080 durante el intervalo activo de la señal SYNC. Esta información se debe preservar almacenándola en retenedores o registros para la generación de señales de control que se suministra a los circuitos lógicos externos.

Las señales de estado se suministran principalmente para el control de los circuitos externos. La simplicidad de interconexión, más que la identificación de los ciclos de máquina, determinan la definición lógica de los bits individuales de estado. Por consiguiente se puede observar que ciertos ciclos de máquina del procesador están identificados en forma única por un sólo bit y otros ciclos no. El bit D6 del estado M1, por ejemplo, identifica claramente un ciclo de obtención de la instrucción (FETCH). El ciclo de lectura de la sección STACK de memoria (STACK READ), en cambio, está indicado por la coincidencia de las señales STACK y MEMR. La información de identificación de los ciclos de máquina también es de gran valor durante las fases de prueba y perfeccionamiento en el desarrollo del sistema. La Tabla 3 lista los bits de estado emitidos durante cada uno de los tipos de ciclo de máquina.

Notas para la Tabla 4:

1. El primer ciclo de memoria (M1) es siempre la obtención de la instrucción (FETCH). Durante este ciclo se obtiene el primero (o único) byte de la instrucción el cual contiene el código de operación.

#### Definición de la Información de Estado

Símbolo	Bit del Canal de Datos	Definición
INTA*	D0	Señal que indica la aceptación de una solicitud de interrupción (INTERRUPT). Esta señal se debe utilizar para transferir la instrucción de recomienzo (RST) al canal de datos cuando DBIN está activa.
WO/	D1	Indica que la operación en el ciclo de máquina corriente será si de una función de escritura en memoria (WRITE) o de salida (OUTPUT) de datos (WO/ = 0). De lo contrario, la operación en ejecución será de lectura de memoria (READ) o de entrada (INPUT) de datos.
STACK	D2	Indica que el canal de direcciones contiene la dirección regresiva recibida del Registro Stack (Stack Pointer).
HLTA	D3	Señal que indica la ejecución de la instrucción HALT.
OUT	D4	Señal para indicar que el canal de direcciones contiene la dirección de un dispositivo de salida y el canal de datos contendrá los datos de salida cuando WR/ esté activo.
M1	D5	Indica que el CPU está en el ciclo de obtención de instrucción (fetch) para leer el primer byte de una instrucción.
INP*	D6	Indica que el canal de direcciones contiene la dirección de un dispositivo de entrada y los datos de entrada deben transferirse al canal de datos cuando DBIN esté activa.
MEMR*	D7	Indica que el canal de datos será utilizado para transferir los datos leídos de la memoria.

\* Estos tres bits de estado se pueden usar para controlar el flujo de información en el canal de datos del 8080.

Tabla 2 Definición de la Información de Estado (Status)

2. Si la señal READY (Listo), determinada por la memoria, es baja durante T2 en cada ciclo de memoria, el procesador entra en el estado TW de espera hasta que se detecte READY alta.
3. Los estados T4 y T5 se presentan, cuando se requieren, para operaciones completamente internas en el CPU. El contenido del canal interno queda disponible en el canal de datos durante T4 y T5 con el propósito de pruebas únicamente. Una "X" indica que el estado (State) está presente pero se utiliza solamente para operaciones internas, tales como la decodificación de la instrucción.
4. Solamente se pueden especificar los pares de registro (rp) B (B y C) o D (D y E).

Tabla de las Palabras de Estado (Status)

		Tipo de Ciclo de Máquina																			
		1	2	3	4	5	6	7	8	9	10										
		INTA	WO/	STACK	HLTA	OUT	M1	INP	MEMR	Instruction Fetch	Memory Read	Memory Write	Stack Read	Stack Write	Input Read	Output Read	Interrupt Write	Halt Acknowledge	Interrupt Acknowledge	While Halt	
00	INTA	0	0	0	0	0	0	0	1	0	1										
01	WO/	1	1	0	1	0	1	0	1	1	1										
02	STACK	0	0	0	1	1	0	0	0	0	0										
03	HLTA	0	0	0	0	0	0	0	0	1	1										
04	OUT	0	0	0	0	0	0	1	0	0	0										
05	M1	1	0	0	0	0	0	0	1	0	1										
06	INP	0	0	0	0	0	1	0	0	0	0										
07	MEMR	1	1	0	1	0	0	0	0	1	0										

**Leyenda:**

- 1 Obtención de la Instrucción
- 2 Lectura de la Memoria
- 3 Escritura en la Memoria
- 4 Lectura del Area Stack de Memoria
- 5 Escritura en el Area Stack de Memoria
- 6 Lectura de Entrada
- 7 Escritura de Salida
- 8 Indicación de una Interrupción Aceptada
- 9 Indicación del Estado de Detenido
- 10 Indicación de una Interrupción Aceptada estando Detenido
- N Palabra de Estado (Status Word)

Tabla 3

5. Estos estados se omiten
6. Sub-ciclo de lectura de memoria; se lee un byte de instrucción o de datos.
7. Sub-ciclo de escritura en memoria.
8. No se requiere la señal READY (listo) durante los sub-ciclos segundo y tercero (M2 y M3). La señal HOLD (suspensión) se acepta durante M2 y M3. La señal SYNC (sincronización) no se genera durante M2 y M3. Durante la ejecución de la instrucción DAD, no se requieren M2 y M3 para la suma del par de registros; no hay acceso a la memoria.
9. Los resultados de estas instrucciones aritméticas, lógicas y de rotación no se transfieren al acumulador (A) hasta ocurrir el estado T2 del próximo ciclo de instrucción. Esto es, el acumulador se carga mientras se obtiene la siguiente instrucción; esta sobreposición de operaciones permiten un procesamiento más rápido.
10. Si el valor de los 4 bits menos significativos del acumulador es mayor que 9, o si el bit de

transferencia auxiliar (Auxiliary Carry) está activo, se suma 6 al acumulador. Si el valor de los 4 bits más significativos del acumulador es ahora mayor que 9 o si el bit de transferencia (Carry) está activo, se suma 6 a los 4 bits más significativos del acumulador.

11. Esta parte representa el primer sub-ciclo (obtención de la instrucción) del próximo ciclo de instrucción.
12. Si la condición especificada se cumple, se envía a las líneas de direcciones A0-A15 el contenido del par de registros WZ en lugar del contenido del contador del programa PC.
13. Si no se cumple la condición especificada, se omiten los sub-ciclos M4 y M5; el procesador prosigue inmediatamente a la obtención de la instrucción (M1) del próximo ciclo de instrucción.
14. Si no se cumple la condición especificada, se omiten los sub-ciclos M2 y M3; el procesador prosigue inmediatamente a la obtención de la instrucción (M1) del próximo ciclo de instrucción.
15. Sub-ciclo de lectura de la memoria Stack.
16. Sub-ciclo de escritura en la memoria Stack.
17. CCC — CONDICION
  - 000 — NZ — Not Zero: Z=0 (No ser cero)
  - 001 — Z — Zero: Z=1 (Ser cero)
  - 010 — NC — Not Carry: CY=0 (Sin Transferencia)
  - 011 — C — Carry: CY=1 (Transferencia)
  - 100 — PO — Parity Odd: P=0 (Paridad Impar)
  - 101 — PE — Parity Even: P=1 (Paridad Par)
  - 110 — P — Plus: S=0 (Signo Positivo)
  - 111 — M — Minus: S=1 (Signo Negativo)
18. Sub-ciclo de Entrada/Salida (I/O): el código de 8 bits de selección del puerto de entrada/salida (I/O Port) se duplica en las líneas de dirección 0-7 (A0-7) y 8-15 (A8-15).
19. Sub-ciclo de salida (Output).
20. El procesador permanece en el estado de detenido (Halt State) hasta que ocurra una interrupción, una reposición o se acepte una solicitud de suspensión (Hold Request). Cuando el procesador acepta una solicitud de suspensión, entra al modo de suspensión, (Hold Mode); cuando este modo termina, el procesador vuelve al estado de detenido. Después de aceptar una reposición (Reset), el procesador comienza a ejecutar la instrucción en la localización cero de memoria. Después de aceptar una interrupción, el procesador ejecuta una instrucción que es

forzada en el canal de datos generalmente llamada instrucción de recomienzo (Restart Instruction)

SSS o DDD	Valor	rp	Valor
A	111	B	00
B	000	D	01
C	001	H	10
D	010	SP	11
E	011		
H	100		
L	101		

## PREFACE

The Sanyo Personal Computer is designed to operate using MSX-BASIC as its language and ROM Cartridges and cassettes are available for various applications. This model is ideal for beginners, wishing to learn about computers, and for those interested in games and children's education, and for all applications both at home and in the office.

This computer includes Operating Instructions and a Programming Manual.

Operating Instructions will explain the connection method of this computer to other equipment.

The Programming Manual in a separate volume provides examples of programming.

### Note:

- 1) These Operating Instructions may not be copied or published either in whole or in part without permission of Sanyo.
- 2) These Operating Instruction may be revised or changed with or without notice.
- 3) Sanyo assumes no liability whatsoever for any claim arising from the use of this computer.

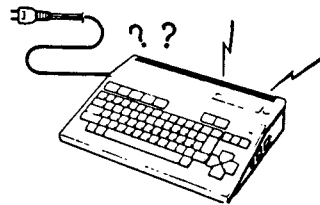
**MSX** is the registered trademark of Microsoft Corp., U.S.A.

## TABLE OF CONTENTS

1. Cautions .....	3
2. Name of Parts .....	5
3. Connections .....	7
• Connecting to TV .....	9
• Connecting to Cassette Tape Recorders .....	9
• Loading ROM Cartridge .....	10
• Connecting to Printer .....	10
• Connecting to Joy Stick .....	11
• Connecting to Light Pen Unit .....	11
4. Functional Check .....	12
5. How to Operate .....	13
• Keyboard .....	13
• Familiarise yourself with the Keyboard .....	18
6. To Save and Load Programs, Use Cassette Tape Recorders .....	22
• To make Cassette Tape Recorder ready .....	22
• Loading program onto computer .....	23
• To save the program .....	25
• Check to see if the program is properly saved .....	26
7. Color TV Adjustment .....	27
8. To Turn off the Power Switch of Computer .....	27
9. Reset Switch .....	28
10. Maintenance .....	28
11. Troubleshooting .....	29
12. Appendix .....	31
• Specifications .....	31
• Terminals Specifications .....	32
• Memory Map .....	34

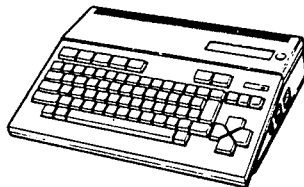
# 1. CAUTIONS

- If, in an unlikely event, any abnormality is noticed with the computer upon turning the power switch on, such as smoke, odd smells, or noise, unplug the power supply at once and contact the dealer from whom you purchased this unit.



- There are no user serviceable parts inside. Never open the cabinet. This will prevent any electric shock, and damage to this unit.

These cautions are indicated on the label stuck at the bottom of the computer cabinet.

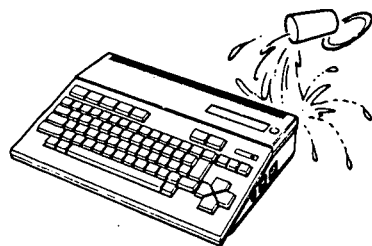


- If the unit fails, send it back to a Sanyo Service Center. Replacement of the power supply cord requires special tools. So do not replace it by yourself for your own safety.



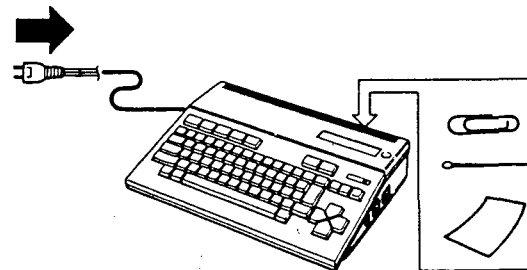
- Avoid locations with high temperature and high humidity, under direct sun light, or with lint or dusts in the atmosphere.

- Keep coffee, tea and soft drinks away from this unit. If any of such liquids are spilled, unplug the power supply, wipe with a soft cloth and at once contact the dealer from whom you purchased this unit.

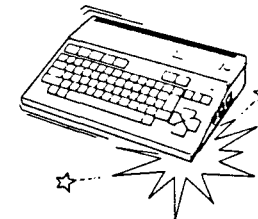


- The maker's name, model No., supply voltage and frequency, and power consumption are indicated on the label stuck at the bottom of the computer cabinet.

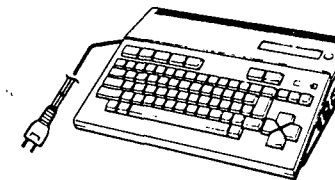
- Do not block the vents provided in the cabinet. Do not insert into the vents any metal object such as pins and clips and any inflammable materials. If any of such metal objects should get into any one of the vents, unplug the power supply at once, and contact the dealer from whom you purchased this unit, as this unit can be damaged when used in such conditions.



- Do not drop, throw or give any abnormal shock to the unit.



- Do not bend the power supply cord with any excessive strength. Do not place any heavy materials on top of cord.

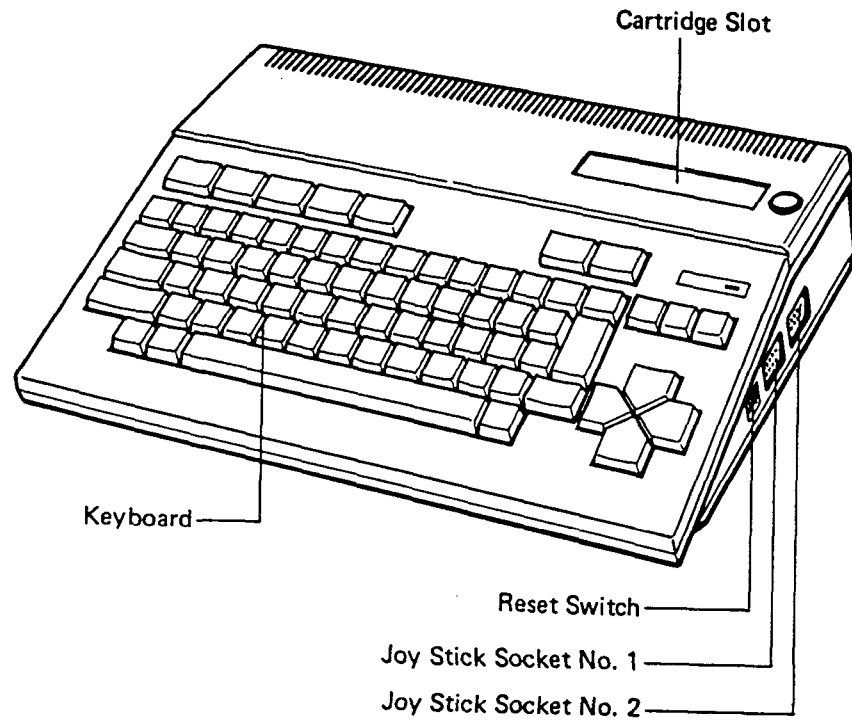


- Unplug the power supply when the unit is not in use for a long period of time.

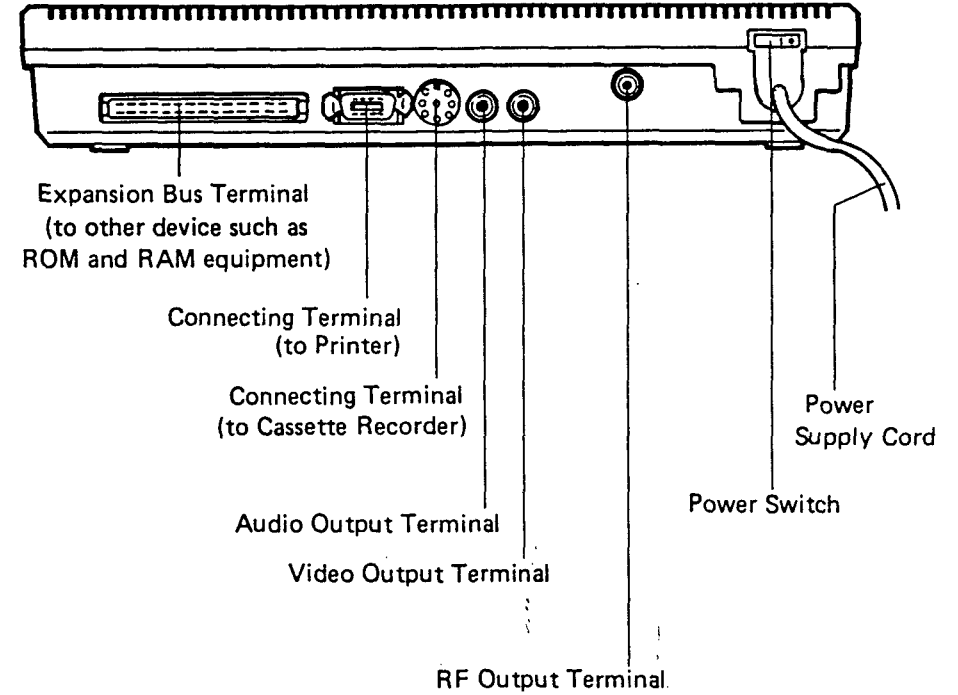


## 2. NAME OF PARTS

### FRONT TOP

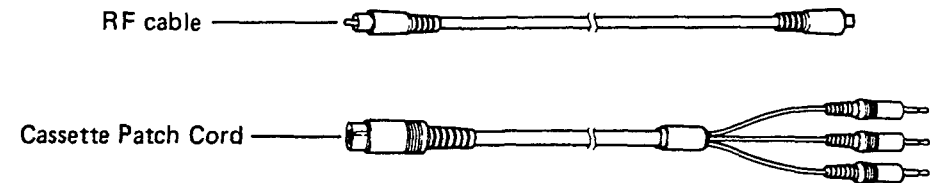


### REAR CABINET



CAUTION: The Expansion Bus is available exclusively for SANYO MSX personal computer.

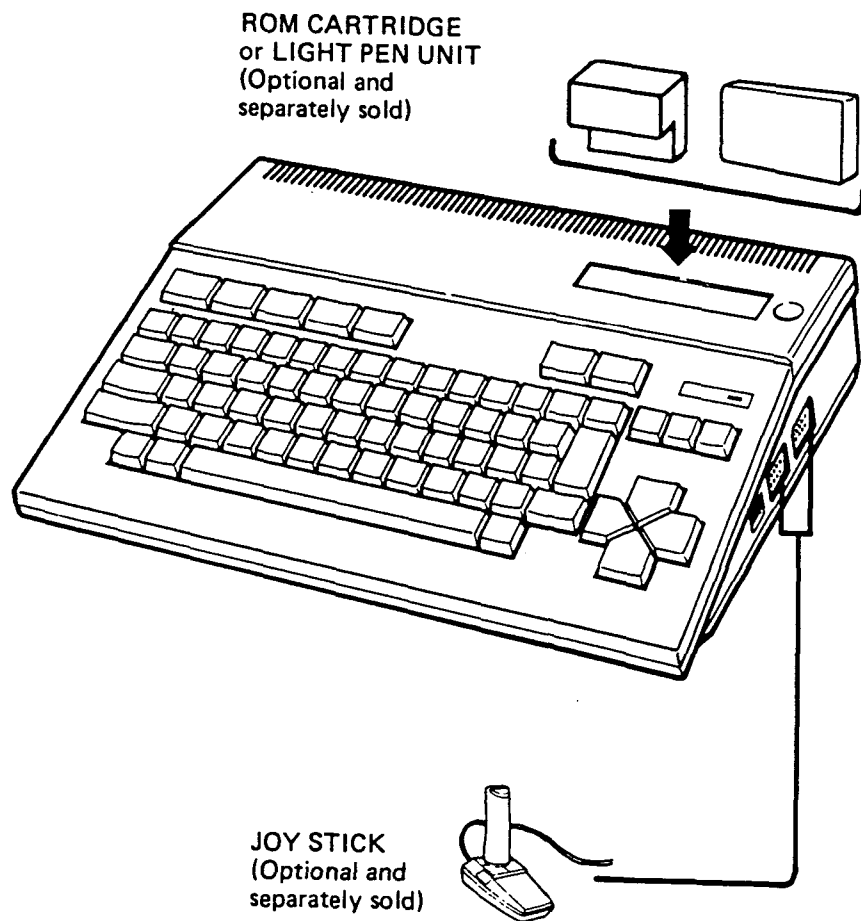
### CONNECTING CABLES (Supplied with the main unit)



# 3. CONNECTIONS

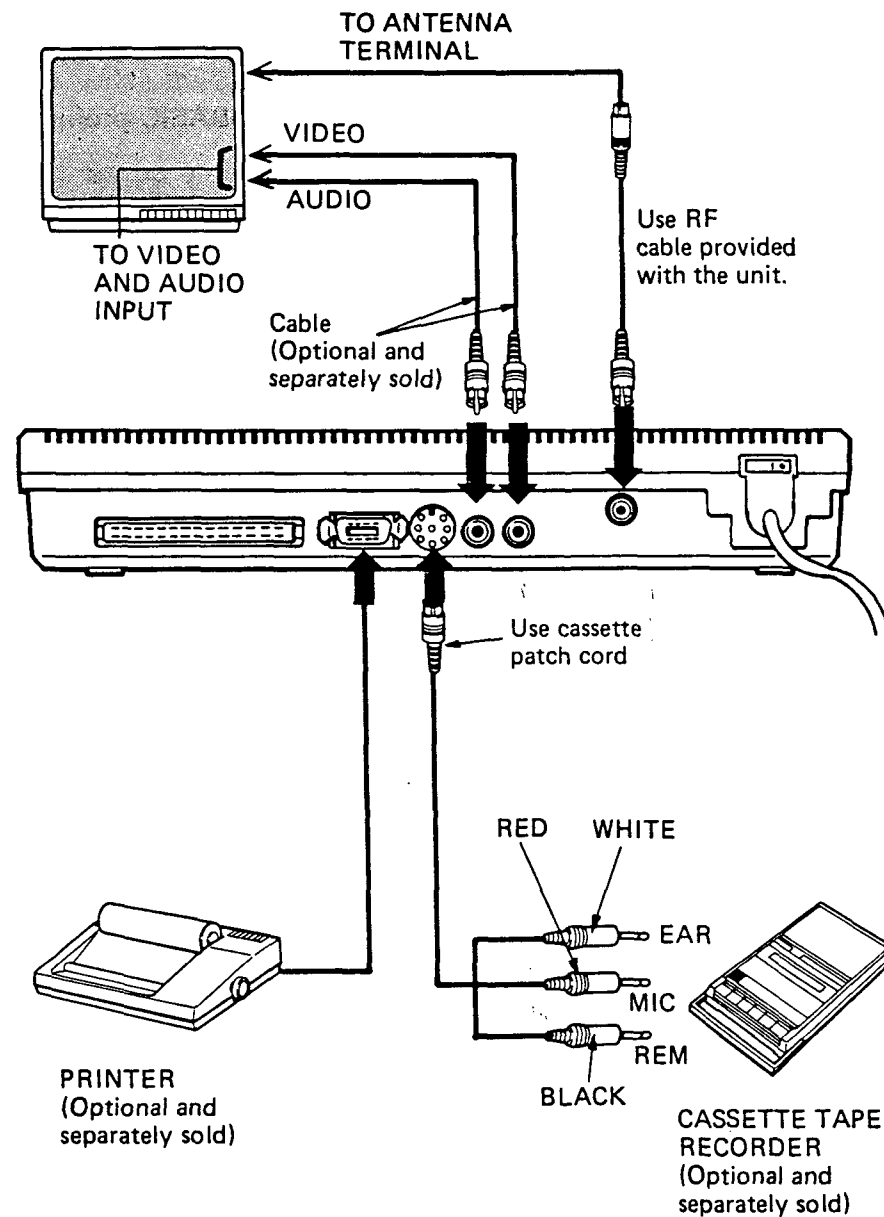
Before connecting to this unit, be sure to turn off power switch of each of the peripheral equipments and this unit.

## TOP AND RIGHT SIDE



## REAR

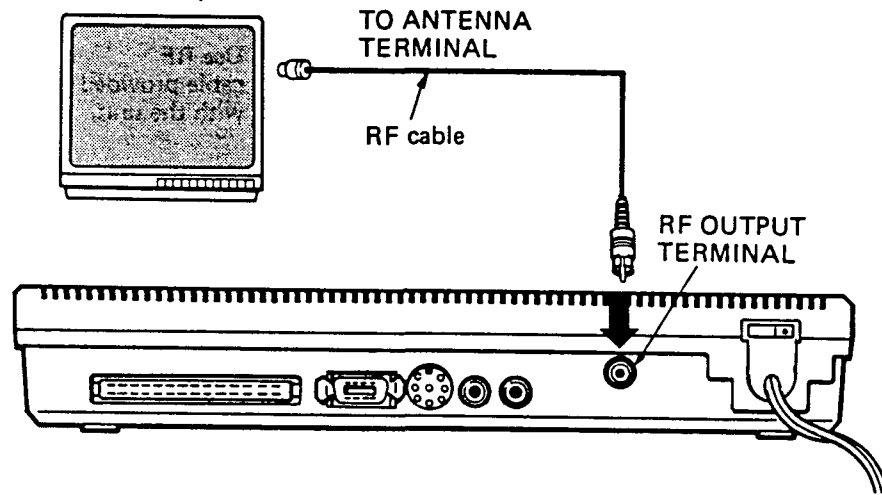
COLOR TV (Optional and separately sold)



### 3-1 CONNECTING TO TV

#### a. Connecting to TV without AUDIO and VIDEO INPUT TERMINALS:

1. Connect RF Output Terminal of computer to Antenna Terminal (75 ohm) of TV Set with RF cable.
2. Set the selected channel of TV until MSX-BASIC prompt appears at top of screen.



#### b. Connecting to TV provided with VIDEO and AUDIO INPUT TERMINALS:

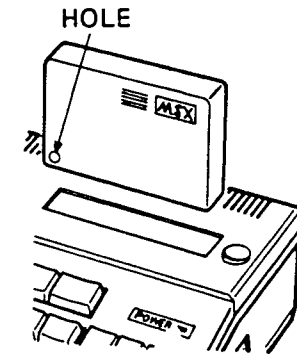
Connect VIDEO and AUDIO OUTPUT terminal of computer to both Video and Audio Input Terminals of TV set with cable.

### 3-2 CONNECTING TO CASSETTE TAPE RECORDERS

Connect this computer to Cassette Tape Recorders with Patch Cords included with unit. If no Remote Control Jack is provided with the Cassette Tape Recorder, do not plug in Remote Jack (Black) with any jack. (In such an instance Start/Stop of Cassette Tape Recorders must be made by hand.)

### 3-3 LOADING ROM CARTRIDGE

Insert a ROM Cartridge (optional and separately sold) into the slot provided at the top right hand corner of cabinet. The lid for the slot needs not be removed as it will give in inside the slot with the insertion of the ROM Cartridge. Make sure that:



1. To set up a ROM cartridge, keep the cartridge hole to the left and insert the cartridge into the slot.
2. The Cartridge is properly locked into position, in other words, push it down all the way, until a click sound is heard. An improper contact may damage the computer.

#### Note:

- This computer will function following the program of the ROM Cartridge. For further details, please consult the Operating Instructions provided with a ROM Cartridge.
- Use only ROM Cartridges bearing the mark of "MSX".

### 3-4 CONNECTING TO PRINTER

Printers based on the Centronics Specifications (optional and separately sold) can be used with this computer. Please consult with your dealer from whom this unit was purchased as to the most suitable printers available.

#### Note:

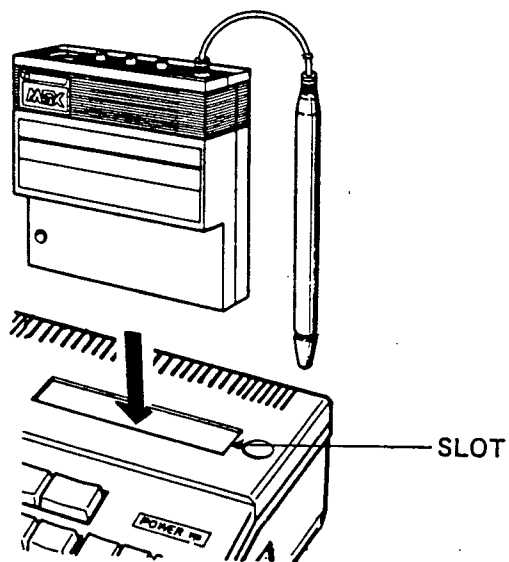
- Use only Connecting Cables having Amphenol type 14-pins.
- The use of an improper connecting cable may result in malfunction.
- Consult also with the Operating Instructions provided with Printer, since certain characters and signs may not be available and thus can be a cause for malfunction.

### 3-5 CONNECTING TO JOY STICK

- Certain Joy Sticks generally available can be used with this computer. Please consult with the dealer from whom the set was purchased for the proper Joy Sticks to be used with the computer.
- Please also consult with the Operating Instructions provided with ROM Cartridge, etc.

### 3-6 CONNECTING TO LIGHT PEN UNIT

- You can draw on the TV screen with a light pen. Light pen unit separately available.
- How to use the light pen is referred to in its accompanying instructions.



## 4. FUNCTIONAL CHECK

1. Confirm all connections.  
Turn off the Power Switch of all equipment connected to the computer. Check to see if all equipment is properly connected as instructed in page 7 to 8.
2. Turn on the Power for all peripheral equipment required for the intended operation.
3. Turn on the Power for the computer.  
About 4 seconds after the Power Switch of the computer is turned on, the copyright message of MSX-BASIC appears successively in 2 frames. If no message appears or the picture on the screen is unstable, depress the RESET button at the right side of the cabinet. If the computer still does not operate properly, read "Troubleshooting" on page 29. Before turning off the Power Switch, please read "To Turn Off Power Switch" on page 27.



```
MSX BASIC version 1.0  
Copyright 1983 by Microsoft  
xxxxx Bytes free  
Ok  
□
```

xxxxx on the screen displays the usable memory expressed in Bytes.

```
color auto goto list run
```

#### Note:

- Before making the initial functional check of your computer, remove ROM Cartridge from the slot, if it is loaded, since the display on the screen may be affected.
- A small portion of the left and right margins of the screen may be missing with some Television Sets.

# 5. HOW TO OPERATE

As soon as the message in the preceding page appears, the computer is ready to accept any input of programs which can be transmitted by either of the following methods:

- Input via Keyboard,
- Input through Cassette Tape, or
- Input through ROM Cartridge or Light pen unit.

First of all, familiarise yourself with the Keyboard.

## 5-1 KEYBOARD

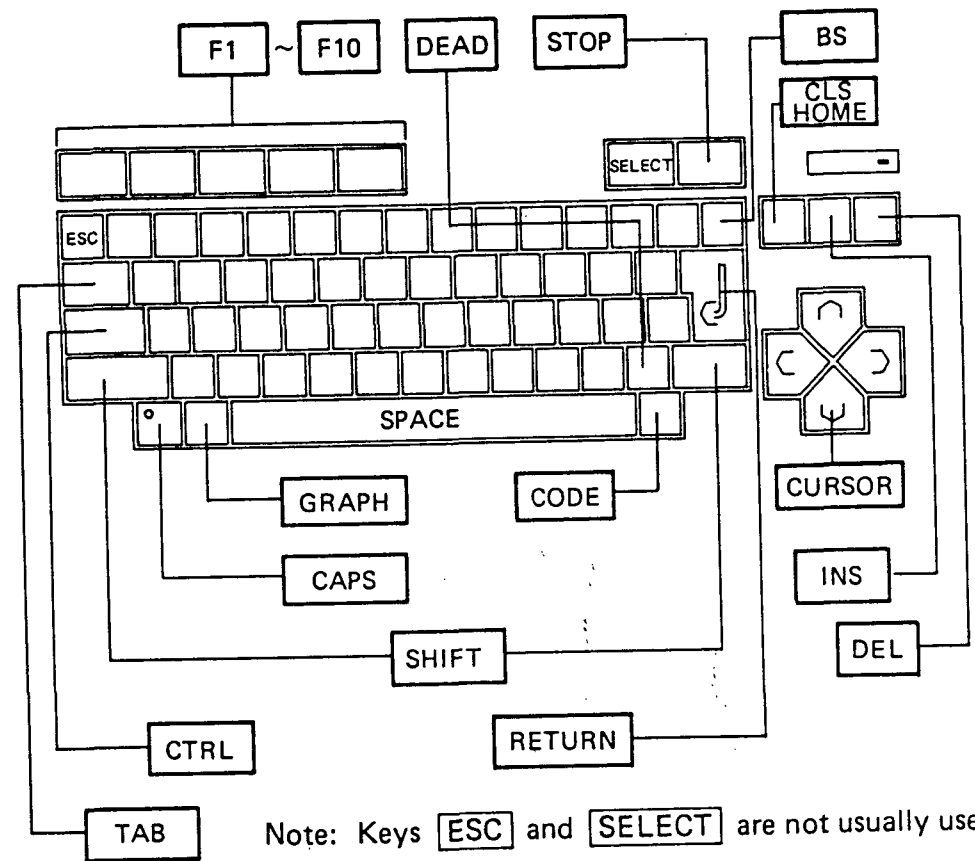
The following characters can be used with the computer through the Keyboard as follows:

Alphabetic Characters	Upper Case: A, B, C, D, ...
	Lower Case: a, b, c, d, ...
Numeric Characters	0, 1, 2, 3, 4, ...
Special Characters	*, /, +, -, " ...
Graphic Characters	▬, ■, ♥, ◆ ...
International Characters	ä, ë, à, á, ... ω, φ, ...

\* Be sure to distinguish between the confusingly similar characters, and numbers:

Numbers	Ⓔ	1	1
Characters	Ⓒ	1	i or I

## KEY NAMES



Note: Keys **ESC** and **SELECT** are not usually used.

## CHARACTERS AND SYMBOLS SET

The SHIFT, CAPS, GRAPH and CODE keys serve also to display the following characters and graphic symbols on the screen.

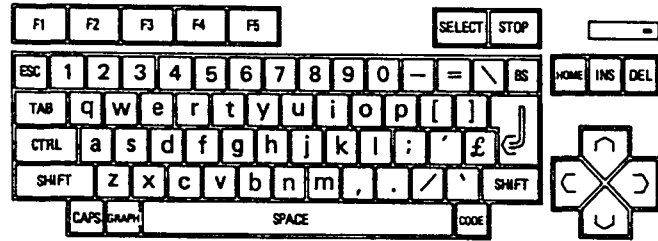
Note:

The mark ⊕ in the key explanation means a toggle key.  
Example: **CAPS** ⊕

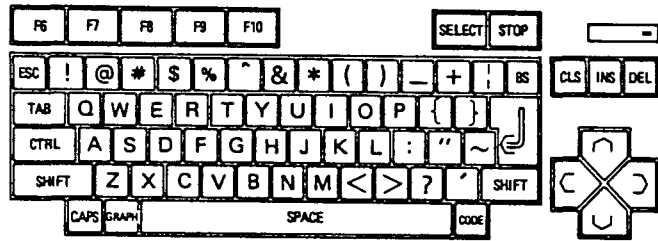
The mark + denotes that one key is pushed down with its preceding key kept depressed.

Example: **SHIFT** + A  
(Push A key while keeping **SHIFT** key down.)

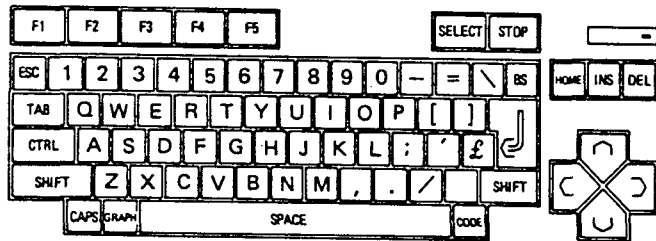
1. NORMAL KEYBOARD



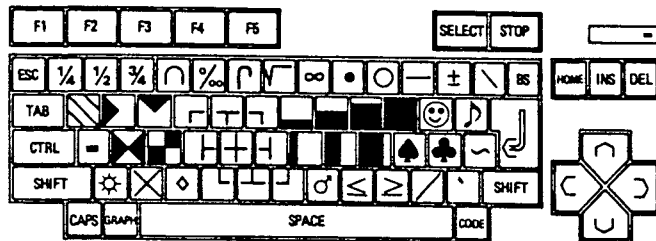
2. SHIFT + ANY KEY



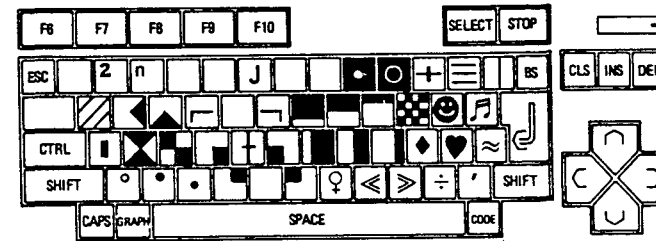
3. CAPS ⊕ ANY KEY



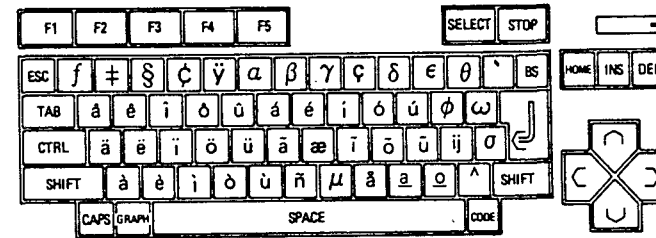
4. GRAPH + ANY KEY



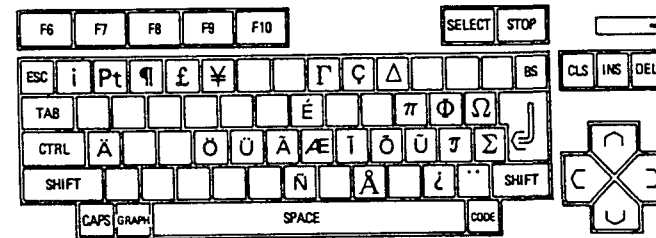
5. GRAPH + SHIFT + ANY KEY



6. CODE + ANY KEY



7. CODE + SHIFT + ANY KEY



## SPECIAL KEYS

**STOP** Key — Used to stop a program running. To interrupt a running, depress this key with the control key being pushed.

**CTRL** Key — Control Key to control key functions. See Programming Manual.

**TAB** Key — Tabulation Key to provide a move of 8 spaces.

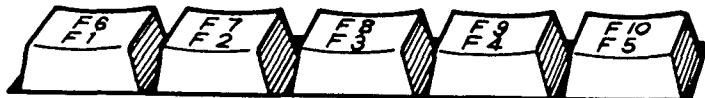
**F1** — **F10** Key — Function Key to command various functions. For further explanation see Note below.

### Note:

This computer has 10 pre-defined Function Keys. The current contents of these Keys are displayed on the last line on the screen and can be redefined by program with KEY statement. The initial values for each Key are:

F1	color	F6	color 15, 4, 4 °
F2	auto	F7	cloud"
F3	goto	F8	cont
F4	list	F9	list.
F5	run	F10	run

(To type f6 through f10, type SHIFT key simultaneously.)



**RETURN** Key — Press this Key to enable the computer to store in its memory all the inputs displayed on the screen when programming.

**DEAD** key — Accents and umlaut can be put on by first depressing one or two corresponding keys of **SHIFT**, **GRAPH** and **CODE** and this key, and then pressing the related alphabetical key a, e, i, o, u or y.

Example	Normal (Dead key only)	.....	è, à
	<b>GRAPH</b> + <b>DEAD</b>	}	
	<b>SHIFT</b> + <b>DEAD</b>		
	<b>SHIFT</b> + <b>GRAPH</b> + <b>DEAD</b>	}	.....
	<b>CORD</b> + <b>DEAD</b>		
	<b>SHIFT</b> + <b>CORD</b> + <b>DEAD</b>		.....
			è, à
			ë, ä

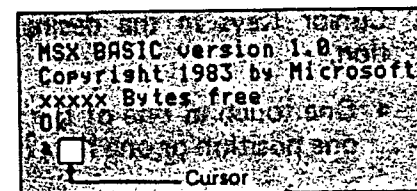
## 5-2 FAMILIARISE YOURSELF WITH THE KEYBOARD

First unload ROM the Cartridge from the cartridge slot of the computer (if it is loaded) and turn the Power Switch off on all Peripheral Equipment and then turn the Power Switch on the computer.

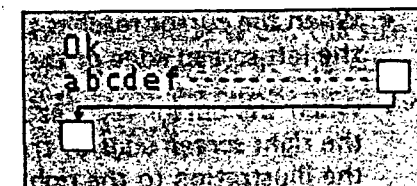
### • TYPE CHARACTERS AND SYMBOLS.

"OK" is now displayed on the last line of the screen.

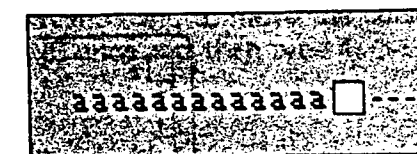
1. Type, for instance the letter "a" and the cursor moves by one position to the right.



2. Keep typing letters until the cursor reaches the extreme right edge, and automatically the cursor is sent to next line.



3. If any Key is kept depressed, the input of the same Key is continuously repeated until the Key is finally released.

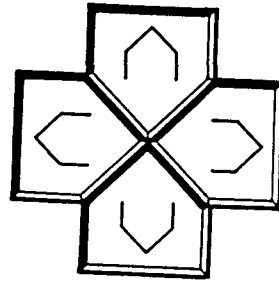


### • TO CLEAR ALL INPUTS ON THE SCREEN:

Type **SHIFT** + **CLS HOME** Keys simultaneously. The cursor will go back to Home Position at the upper left corner of the screen.

• TO ADD, CHANGE AND ERASE ALL TYPES OF CHARACTERS:

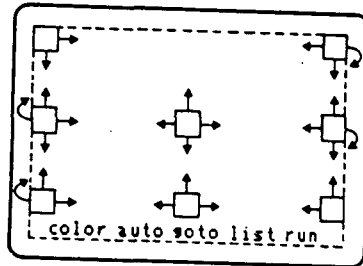
Typing errors made during the program input can be easily corrected. To make a partial correction, move the cursor to the location where correction is necessary.



1. To move the cursor

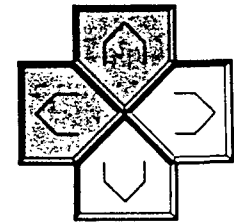
To move the cursor, use one of the Cursor Keys in the desired direction.

- One touch at one of the Cursor Keys will move the character by one position or one line, up, down, left or right as desired.
- The Cursor Key will keep the cursor travelling in the direction of the arrow, so long as it is kept depressed.
- When the cursor reaches the right screen edge it will move over to the left screen edge of the next line.
- When the cursor reaches the left screen edge, it will move over to the right screen edge of the penultimate line. For more details see the illustration to the right.



Note:

- The cursor travels only within the width of the display set in text mode by a statement WIDTH.
- The cursor will not travel over the last line of the display where the meanings of each Function Key are displayed.
- **CLS HOME** will move the cursor to the upper left corner of the screen, or Home Position.
- The cursor travels also in any diagonal direction as illustrated to play game programs, etc.



2. To insert characters:

**INS** ⊕ or **CTRL** + **R** :

Toggle switch for insert mode. When insert mode is on, the size of the cursor is reduced and characters are inserted at the current "Cursor" position. Characters to the right of the cursor move right as new ones are inserted. Line wrap is observed.

Example:

0 1 2 3 4 5 6  
                   ↑  
                   Cursor

To add 4 between 3 and 5, move the cursor over 5. Type INS once and then type 4.

3. To change characters:

When insert mode is off, the cursor is returned to the normal size and the typed characters will replace existing characters.

Example:

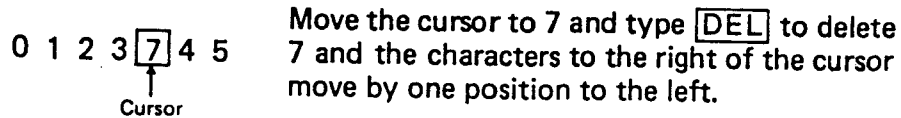
0 1 2 3 4 5 6  
                   |  
                   Cursor

To change 7 to 4, move the cursor to 7 and type 4 over 7.



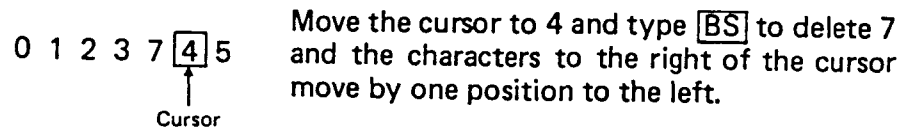
4. Type **DEL** to delete characters:

Example:



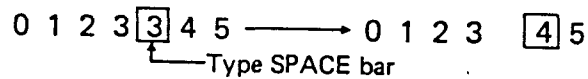
5. Type **BS** to delete the character to the left of the cursor:

Example:

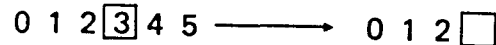


6. Type **SPACE** bar to delete the character without moving the characters to the right of the cursor.

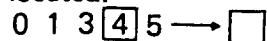
Example



7. Type **CTRL** + **E** to delete all characters to the right of the cursor.



8. Type **CTRL** + **U** to delete the whole line where the cursor is located.



## 6. TO SAVE AND LOAD PROGRAMS ... USE CASSETTE TAPE RECORDERS

Programs can be saved and loaded using regular Cassette Tape Recorders. Both monaural and stereo are equally fine. One with a remote control jack is preferable since it will permit an automatic operation of cassette tape recorders with the computer command. To avoid errors in the process of saving and loading, however, the use of a Sanyo Data Recorder, which is specifically designed for DATA saving, is recommended.

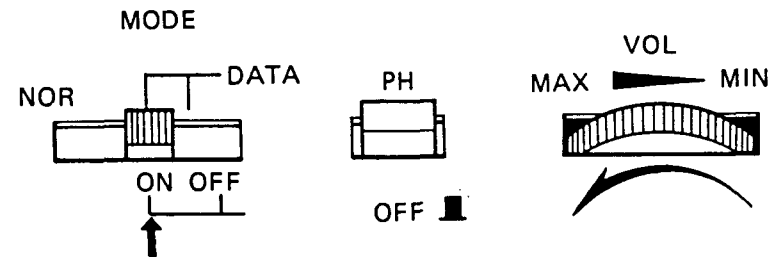
The importance of this DATA saving function can be well recognized because all the inputs memorized in RAM of the computer are lost the moment the power is turned off, while some of the programs may take hours of hard work for inputs, debuggings, and improvements.

The saving and loading functions of the computer permit the use of programs made by both yourself and others.

### 6-1 TO MAKE CASSETTE TAPE RECORDER READY

- When the Sanyo Data Recorder is used:

1. Set the thumb wheel volume VOL knob at the maximum position by turning it all the way to the left. (Otherwise, saving failure may result.)



2. Turn off the PH phase switch by setting the push button at HIGH position.
3. Set the slide switch to DATA mode, either to the ON (for monitor) or OFF (without monitor), whichever is required. Set the Mode switch to the DATA ON position.

• When a Cassette Tape Recorder other than Data Recorder is used.

1. Set the volume control knob around the mid position, neither too low nor too high.
2. Set the tone control knob (if provided) around the mid position, neither too low nor too high.

## 6-2 LOADING PROGRAM ONTO COMPUTER

1. Place the cassette tape in the cassette tape recorder.
2. Rewind the cassette all the way to the starting position.
3. Depress the PLAY key. (If the Cassette Tape Recorder is not provided with the remote jack, this STEP must follow after the STEP 4 below.)
4. Type CLOAD "DEF" if the program is to be named DEF, and then type return key.
5. With a small "click" sound, the tape starts to run to search for the file name of the commanded program.

6. **FOUND:** will be displayed on the screen. In this search mode, any file other than the one commanded will be identified on the screen as **Skip** followed by the file name.

Example: If the program with the file name of DEF is followed by another one with the file name of ABC, the screen will display:

```

cload "DEF" ← Type input code
Skip : ABC ← In search mode
Found: DEF ← Upon locating the file
Ok
□ ← Cursor

```

7. Any program saved by means of **CSAVE** can be loaded by typing only **CLOAD**, (Note the quotation mark " must not follow **CLOAD**) if the file name is forgotten or unknown. All preceding file names will be found and loaded one after another by repeating **CLOAD**. Any file programmed in the machine language, saved under **BSAVE** command, cannot be loaded without typing the exact file name.
8. To discontinue search mode, after **CLOAD** is typed in, type **CTRL** + **STOP** simultaneously.
9. Adjust the volume level of the Cassette Tape Recorder between the mid to low range as necessary, if at first attempt at **CLOAD**, the loading is not successful.

## 6-3 TO SAVE THE PROGRAM

As explained earlier in these Operating Instructions, the program memorized in the RAM of the computer can be erased easily if:

- a. Power Switch is turned off,
- b. Reset Button is depressed once, or
- c. NEW command is typed in.

To save the program, follow the steps specified hereunder:

1. Set a blank cassette tape (other than the tapes provided with this computer) into the Cassette Tape Recorder in position.
2. Set it to recording mode by depressing PLAY and RECORD buttons, or as otherwise instructed in the Operating Instructions of the Cassette Tape Recorder. (If the cassette Tape Recorder is not provided with the remote jack, this STEP must follow after the STEP 4 below.)
3. Type **CSAVE "MSX"** if the program is to be named MSX, or any other file name desired, and then press the return key ().
4. With a small "click" sound, the tape starts to run to save the program.
5. **Ok** will be displayed on the screen and the cassette will stop running. (If the Cassette Tape Recorder is not provided with remote jack, stop it manually when **Ok** appears.)

```
CSAVE "MSX"  
Ok  

```

6. To make an additional back up file, type as follows: In this example, when the program with the file name of "DEMO" is found, and a blank cassette is replaced with the cassette containing "DEMO", the file "DEMO" will then be saved on the blank cassette under the file name of "MSX").

```
cload "DEMO"  
Found : DEMO  
Ok  
csave "MSX"  
Ok  

```

## 6-4 CHECK TO SEE IF THE PROGRAM IS PROPERLY SAVED:

While the program is still stored in the RAM of the computer, it is possible to check whether the program is properly saved by giving **CLOAD?** command as follows:

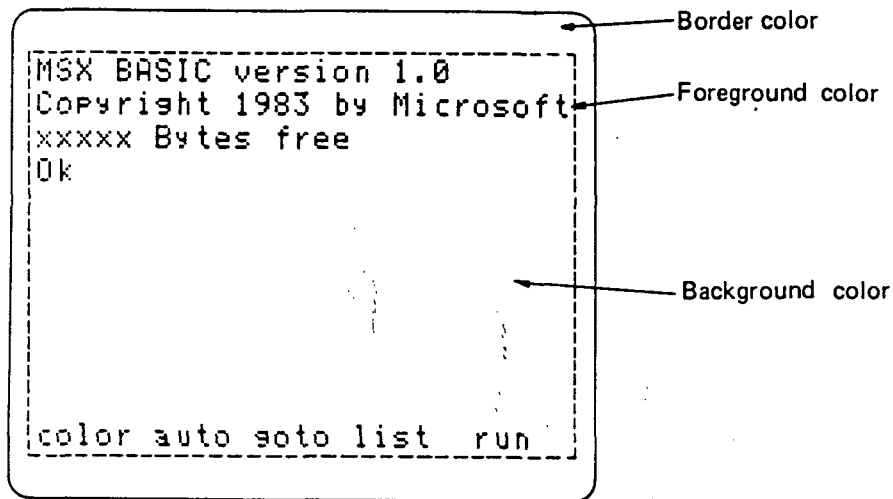
1. Rewind the cassette tape to the position where saving for the file was started.
2. Suppose the file name just saved is "FLOWER", Type **CLOAD? "FLOWER"** and . Please note that up to six characters can be used for the file name, excepting colon (:) and all other characters identified under character code 0 and 255. See Programming Manual for further details.
3. Depress the PLAY button of the Cassette Tape Recorder. (If the Cassette Tape Recorder is not provided with the remote jack, this STEP must follow after the STEP 4 below.)
4. Type . The cassette tape starts running and the programs stored in memory of the computer and on the cassette tape are compared all the way step by step.
5. **Ok** will appear on the display if the program is properly saved, and **Verify error** if any difference is noted. If the latter is the case, repeat the **CSAVE** process one more time by adjusting the volume level of the Cassette Tape Recorder either up or down slightly.

**Remarks:** The user of the computer should not be frightened with any Error message which is caused by input errors and which appears on the display since no error can damage or destroy the computer. If the input is corrected, it will resume functioning again. See Programming Manual for more details.

## 7. COLOR TV ADJUSTMENT

Color quality can be affected greatly by how the tint and color of Color TV set are adjusted. Use the color pattern and adjust the tint and color to the optimum settings.

When the power switch is turned on, the background and the border are automatically set in dark blue and the foreground (alphanumeric characters) in white.



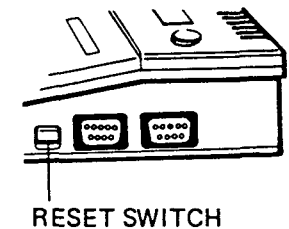
## 8. TO TURN OFF THE POWER SWITCH OF THE COMPUTER

- 1) First save the program stored in RAM memory of the computer, as the moment the power switch is turned off, it will be lost completely.
- 2) Turn off the power switch of the computer, before all other power switches of peripheral equipment are turned off.
- 3) Do not repeat on/off operations of the power switches of computer and its peripheral equipment at short, quick intervals as it may result in malfunctions.

## 9. RESET SWITCH

The reset switch is located at the right side of the cabinet. One push of this reset switch will result in:

- 1) The loss of all memory of the program stored in RAM of the computer.
- 2) The same copyright statement (as is displayed on the screen when the power switch is first turned on) appears, if not connected to other devices, such as ROM and RAM Cartridges.
- 3) If connected to other devices, such as ROM and RAM Cartridge, the initial position is resumed.



## 10. MAINTENANCE

- Use soft cloth to wipe stains and dust off the cabinet of the computer.
- Use soft cloth moistened with detergent for removing the stubborn stains and then wipe the surface with dry soft cloth.
- Do not use thinners, benzenes, and other chemical solvents.

# 11. TROUBLESHOOTING

If by any chance your computer should fail to operate, go through the following check list and make your own diagnosis.

## NO PICTURE:

- a. Check to see if the power supply pilot lamp is lit. If not, check the connection to the power outlet.
- b. Check to see if the power is turned on for your television set.
- c. Check to see if the video output or RF output connections are correctly made.
- d. Check to see if a ROM cartridge (if loaded) is properly loaded. Unload it first and load it again.
- e. Check to see if the brightness and contrast controls are set at optimum positions (for RF connection).

## THE PICTURE ROUS, OR INPUT SIGNAL IS WEAK.

- a. Check to see the input signal channel of the television set correctly mached to the output signal channel of the computer. (for RF connection)
- b. Adjust the fine tuning control of the television set. (for RF connection)
- c. Adjust horizontal and vertical holds of the television set.
- d. Check to see if the video connecting cable is correctly plugged in to the television set.

## ROM CARTRIDGE FAILS TO OPERATE.

- a. Check to see if the ROM cartridge is locked into position ALL the way, correctly.
- b. Check to see if the terminals of the ROM cartridge are clean.
- c. Press the Reset Button at the right side of cabinet.

## COMPUTER CEASED TO OPER- ATE DURING USE.

- a. Power could have been temporarily cut off for a brief moment.
- b. Noise from other electrical equipment could have entered the computer.
- c. Check to see if the peripheral equipment such as printers and cassette tape recorders are correctly connected to the computer.
- d. Check to see if there is no error in the program.

## LOADING OR SAVING USING CASSETTE TAPE RECORDER FAILS.

- a. Check to see if the patch cord is correctly connected at both ends.
- b. Check to see if the operation is correctly made as specified in Section 6.
- c. Check to see if the batteries are fresh and not worn out.
- d. Adjust the tone and the volume levels of cassette tape recorder.
- e. Loading from the cassette tape saved in 2400 baud can only be made with cassette tape recorders designed for DATA processing such as Sanyo Data Recorder.

# 12. APPENDIX

## 12-1 SPECIFICATION

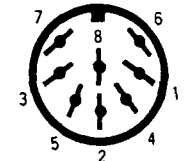
- CPU ..... Z-80A Compatible (3.58 MHz)
- Memory ..... MAIN MEMORY: 64 KB RAM  
SYSTEM MEMORY: 32 KB ROM  
VIDEO MEMORY: 16 KB RAM
- Display ..... Text I : 24 Lines x 40 Characters, 16 colors  
Text II : 24 Lines x 32 Characters, 16 colors  
Graphic : 256 x 192 dots, 16 colors  
Multi Color : 64 x 48 blocks, 16 colors
- Sound ..... 8 octaves, Triple chord output
- Keyboard ..... 73 Keys
- Interface ..... Printer : 8 bit parallel  
Cassette : FSK method, 1,200/2,400 bauds  
Joystick : 2 slots  
Cartridge Bus : 1 slot, 50 pin  
Expansion Bus: 1 slot, 50 pin
- Output signal ..... Video 1.0 VP-P, 75 ohms, RCA phono jack  
Audio 1.26 VP-P, 600 ohms, RCA phono jack  
RF (UHF CH36, 591.25 MHz) RCA phono-DIN 75 ohms plug
- Dimensions ..... Cabinet : W385 x H62 x D242 (mm)
- Weight ..... Net weight : 2.2 Kg
- Power Supply ..... 220 ~ 240 Volts, AC 50/60 Hz
- Power Consumption ..... 11 watts

### ACCESSORIES

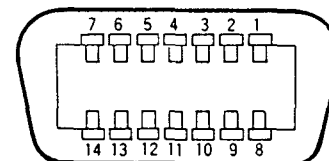
- RF cable
- Patch Cord for connecting to cassette tape recorder
- Operating Instructions
- Programming Manual
- Cassette Tape . . . 2 pcs. (MPC-100 ONLY)

## 12-2 TERMINALS SPECIFICATIONS

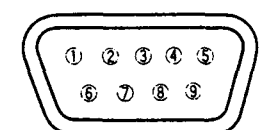
### • Cassette Tape Recorder Terminal

Pin No.	Name		Pin Connection
1	GND	-	
2	GND	-	
3	GND	-	
4	CMTOUT	Output	
5	CMTIN	Input	
6	REM+	Output	
7	REM-	Output	
8	GND	-	

### • Printer Terminal

Pin No.	Name	Pin Connection
1	PSTB	
2	PDB0	
3	PDB1	
4	PDB2	
5	PDB3	
6	PDB4	
7	PDB5	
8	PDB6	
9	PDB7	
10	NC	
11	BUSY	
12	NC	
13	NC	
14	GND	

### • Joy Stick Terminal

Pin No.	Name		Pin Connection
1	FWD	Input	
2	BACK	Input	
3	LEFT	Input	
4	RIGHT	Input	
5	+5V	-	
6	TRG1	Output	
7	TRG2	Output	
8	OUT	Output	
9	GND	-	

# Puntos de entrada relacionados con el manejo de cartuchos

Las BIOS descritas en este tema soportan el sistema de cartuchos. (Véase la parte segunda, tema 50, "Memoria y cartucho del MSX".)

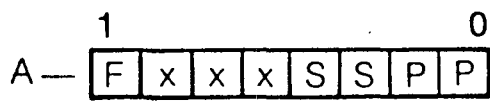
**RDSLTL      000C**

*CALL &H000C*

Esta llamada se emplea para leer una posición de un cartucho.

## PARAMETROS DE ENTRADA

El par de registros HL han de contener la dirección de la posición a leer, y el acumulador indicará de qué cartucho se lee. El contenido del acumulador tiene el siguiente significado: los dos bits menos significativos indicarán el cartucho primario a emplear (0-3); los dos bits siguientes indicarán el cartucho secundario (0-3); los tres bits que siguen no tienen significado alguno; el bit más significativo ha de estar a 1 para emplear el cartucho secundario (en caso contrario se empleará el primario).



- Número del cartucho primario (0-3)
- Número del cartucho secundario (0-3)
- No se emplean
- 1: si se emplea el cartucho secundario
- 0: cartucho primario

## PARAMETROS DE SALIDA

En el acumulador estará el contenido de la posición especificada.

## REGISTROS Y POSICIONES DE MEMORIA AFECTADOS

Altera el contenido de los registros AF, BC y DE.

*Nota:* Esta rutina desactiva todo tipo de interrupciones. Si intentas acceder a la página 3 de un cartucho harás que el ordenador se bloquee. (Véase ENASLT para evitarlo.)

## WRSLT 0014

Ejecución a través de CALL&H0014.

Se emplea para escribir en una posición determinada de un cartucho.

## PARAMETROS DE ENTRADA

El par de registros HL contendrán la dirección de la posición a escribir del cartucho; el acumulador especificará el cartucho a emplear y el registro E tendrá el valor del byte a escribir. El contenido del acumulador tendrá el mismo formato que en la rutina RDSLT (000C).

## PARAMETROS DE SALIDA

Esta rutina no devuelve parámetros.

## REGISTROS Y POSICIONES DE MEMORIA AFECTADOS

Con esta llamada se alterarán los contenidos de los registros AF, BC y D.

*Nota:* La rutina descrita desactiva todo tipo de interrupciones. Si intentas acceder a la página 3 del cartucho se te bloqueará el ordenador. (Véase ENASLT para evitarlo.)

## CALSLT 001C

Ejecución a través de CALL&H001C.

Esta rutina hace una llamada a un cartucho, que selecciona una página de otro cartucho y hace una llamada a una determinada dirección.

## PARAMETROS DE ENTRADA

El registro IX contendrá la dirección a llamar, y el byte más significativo del registro de índice IY especificará el cartucho. El formato de este byte de IY es el mismo que el del acumulador en la RDSLT. Los parámetros a pasar pueden estar en los registros AF, BC, DE y HL, pero nunca en los registros alternativos.

## PARAMETROS DE SALIDA

No se devuelve ningún parámetro, excepto si son devueltos por la llamada del cartucho; estos parámetros podrán devolverse en cualquier registro excepto en el acumulador alternativo, ya que éste contendrá el byte de selección de cartucho antes de la segunda llamada.



## REGISTROS Y POSICIONES DE MEMORIA AFECTADOS

El contenido de los registros AF', BC', DE' y HL' se cambiará antes de ejecutar la llamada al cartucho interno.

*Nota:* Esta rutina desactiva las interrupciones. Se podrá ejecutar a través de la instrucción RST6. (Véase CALLF, tema 29.) Si intentas acceder a la página 3 de un cartucho se te bloqueará el ordenador. (Véase ENASLT para evitarlo.)

## ENASLT            0024

Ejecución a través de CALL&H0024

Esta llamada selecciona una página de un cartucho.

### PARAMETROS DE ENTRADA

Los dos bits más significativos del registro H se emplean para seleccionar la página adecuada, del siguiente modo:

BMS	Página seleccionada
00	0000-3FFF
01	4000-FFFF
10	8000-BFFF
11	C000-FFFF

El acumulador indica el cartucho a seleccionar, según se explicó en RDSLTL.

### PARAMETROS DE SALIDA

No devuelve ningún parámetro.

## REGISTROS Y POSICIONES DE MEMORIA AFECTADOS

Esta llamada altera los contenidos de AF, BC, DE y HL.

*Nota:* Esta rutina desactiva todo tipo de interrupciones.

Con esta llamada se evitarán los problemas de acceso a la página 3 en las instrucciones RDSLTL, WRSLTL y CALSLTL. Por ejemplo, si quieres leer la dirección &HD000 en la página 3, debes emplear los siguientes códigos:

CALL	&H0138	; LEE EL CONTENIDO DEL REGISTRO DE SELECCION DEL CARTUCHO PRIMARIO.
PUSH	AF	; APILA DICHO REGISTRO.
LD	HL;&HD000	; DIRECCION A LEER.
PUSH	HL	; APILA DICHA DIRECCION.
LD	A,3	
DI		; DESACTIVA LAS INTERRUPCIONES, YA QUE ESTAS AFECTAN A LA PAGINA 3.
CALL	&H0024	; SELECCIONA LA PAGINA 3.
POP	HL	; DESAPILA LA DIRECCION.

LD	H,(HL)	; TOMA EL CONTENIDO DESEADO.
POP	AF	
CALL	&H013B	; VUELVE A ACTIVAR LA PAGINA 3 DEL SISTEMA.
EI		
LD	A,H	; DEVUELVE EL VALOR DESEADO EN EL ACUMULADOR.
RET		

Se pueden emplear otros métodos parecidos para escribir y llamar subrutinas de la página 3 de otro cartucho.

## **RSLREG            0138**

Ejecución a través &H0138.

Esta llamada lee el contenido del registro de selección del cartucho primario.

### **PARAMETROS DE ENTRADA**

No necesita de estos parámetros.

### **PARAMETROS DE SALIDA**

El acumulador contendrá una copia del registro de selección del cartucho primario.

### **REGISTROS Y POSICIONES DE MEMORIA AFECTADOS**

Sólo se cambia el valor contenido en el acumulador.

*Nota:* Esta rutina es una simple instrucción IN (IN&HA8 posiblemente), seguida de RET.

## **WSLREG            013B**

Ejecución a través de CALL&H013B.

Con esta llamada escribimos en el registro de selección del cartucho primario.

### **PARAMETROS DE ENTRADA**

El acumulador ha de tener el valor a escribir.

### **PARAMETROS DE SALIDA**

Esta rutina no devuelve ningún parámetro.

### **REGISTROS Y POSICIONES DE MEMORIA AFECTADOS**

No afecta a ningún registro, ni tampoco a posición alguna.

*Nota:* Esta rutina es sencillamente una instrucción OUT (OUT&HA8 probablemente), seguida de RET.

## **CALBAS            0159**

Ejecución a través de CALL&0159.

---

El intérprete de BASIC emplea este punto de entrada para llamar a una extensión de él mismo, contenida en un cartucho.

#### **PARAMETROS DE ENTRADA**

El registro IX ha de contener la dirección de llamada.

#### **PARAMETROS DE SALIDA**

La salida producida depende de la llamada al cartucho.

#### **REGISTRO Y POSICIONES DE MEMORIA AFECTADOS**

También dependen de la llamada.

PRESUPUESTO

ANALIZADOR DE ESPECTRO DE AUDIO.-

Hoja núm. 1

Pap. Babón

N <sup>o</sup> PIEZAS	CONCEPTOS	IMPORTES	
		PARCIALES Pesetas	TOTALES Pesetas
240	Resistencias del 2%	20	4800
34	Resistencias del 10%	5	160
2	Potenciómetros log. 100K	100	200
30	Potenciómetros ajust. 100K	35	1050
44	Condensadores electrolíticos	40	1760
52	Condensadores planos	20	1040
335	Diodos Leds	25	8290
40	Diodos 1N4148	10	400
8	Diodos 1N4001	10	80
31	Integrados TL084	400	12400
3	Transistores BC517	55	165
1	Transistor BC547B	35	35
30	Transistores BC516	55	1650
2	Integrados 4067B	850	1700
2	Integrados 4515B	400	800
1	Integrado 4060B	225	225
3	Integrados 4070B	100	300
1	Integrado 78L05	400	400
1	Integrado 4010	125	125
4	Integrados 4015	150	600
1	Integrado 4070	100	100
1	Integrado 7812	175	175
1	Integrado 7908	175	175
1	Integrado 7912	175	175
1	Integrado LF356	300	300
130	Condensadores del 2'5%		

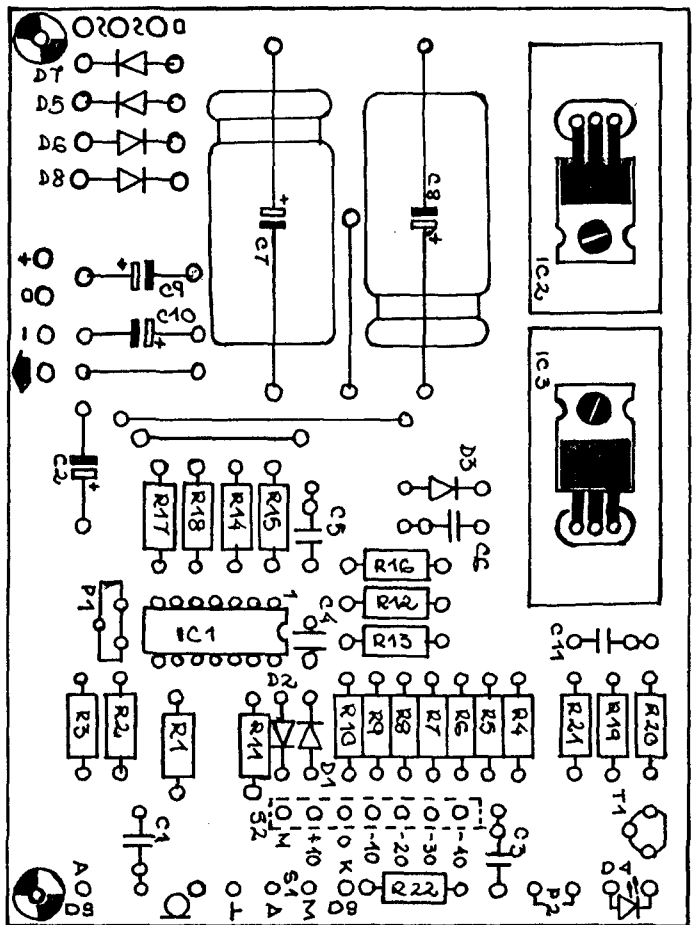
N <sup>o</sup>	CONCEPTOS	IMPORTES	
		PARCIALES Pesetas	TOTALES Pesetas
25	Condensadores del 5%		
1	Interruptor monopolar	210	210
1	Interruptor 6 posiciones	327	327
1	Interruptor bipolar de red	225	225
1	Pulsador inversor	197	197
1	Pulsador	44	44
1	Cápsula microfónica KE-4-211-2	191	191
1	Conexión hembra estéreo 6'3mm.	119	119
2	Disipadores de calor 10°C/W	300	600
2	Disipadores de calor 17°C/W	300	600
1	Fusible 0'5A con portafusible	200	200
1	Transformador de red 2.10/15v.	1000	1000
1	Integrado ZN425E	700	700
TOTAL DEL PRESUPUESTO.....			=====
			41518
ASCIENDE EL PRESENTE PRESUPUESTO A LA CANTIDAD DE: 41518 PESETAS.			

## INDICE DE PLANTILLAS

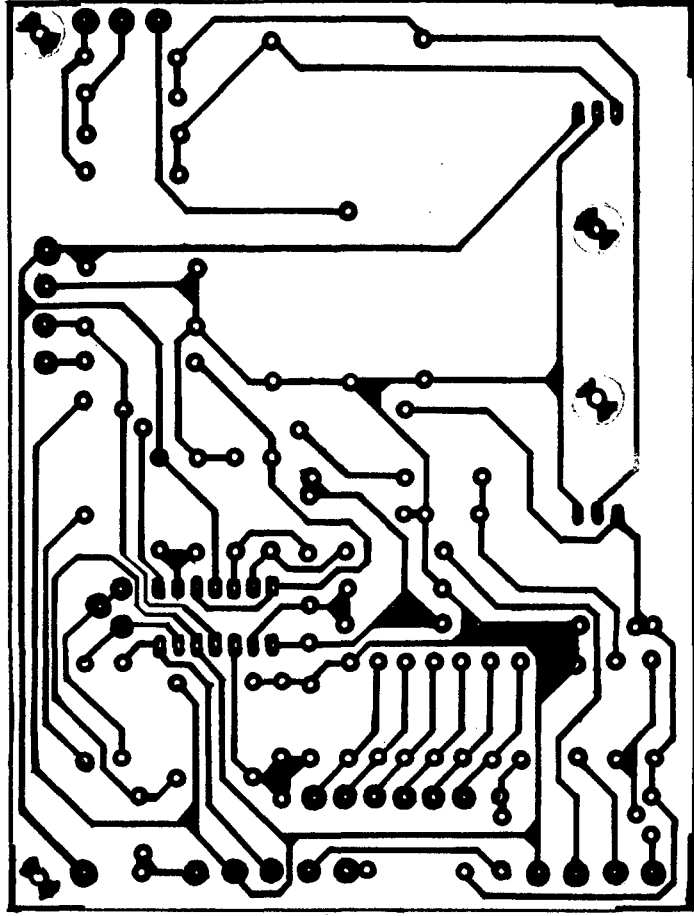
- 1.- PLACA DE ENTRADA Y ALIMENTACION.
- 2.- PLACA DE FILTRO.
- 3.- PLACA DE VISUALIZACION.
- 4.- PLACA DE RUIDO ROSA.
- 5.- PLACA BASE.
- 6.- ALGUNOS EJEMPLOS DE DISEÑO DE PLACAS EN PA  
PEL MILIMETRADO.

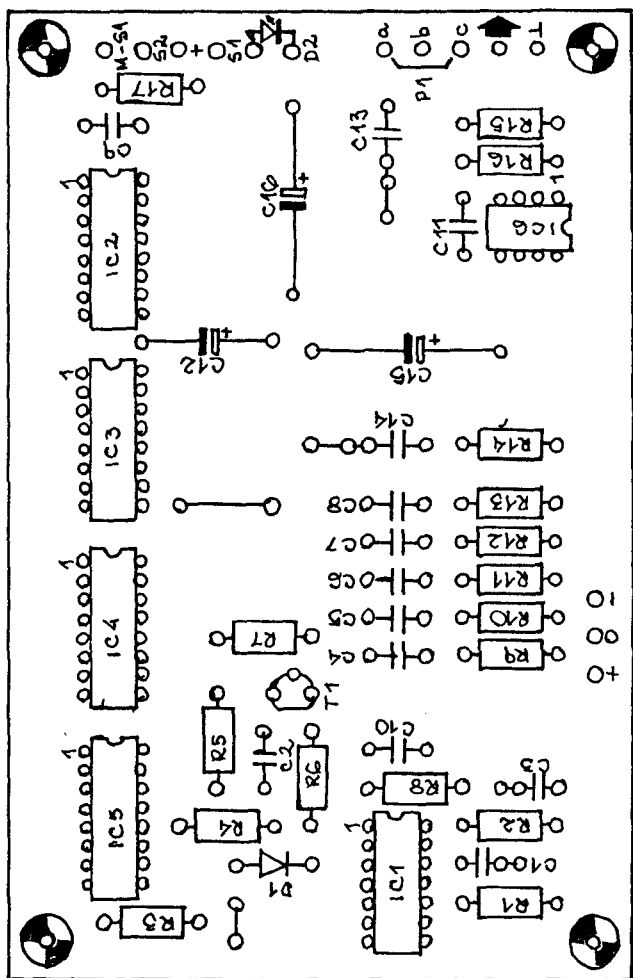
## INDICE DE PLANOS.

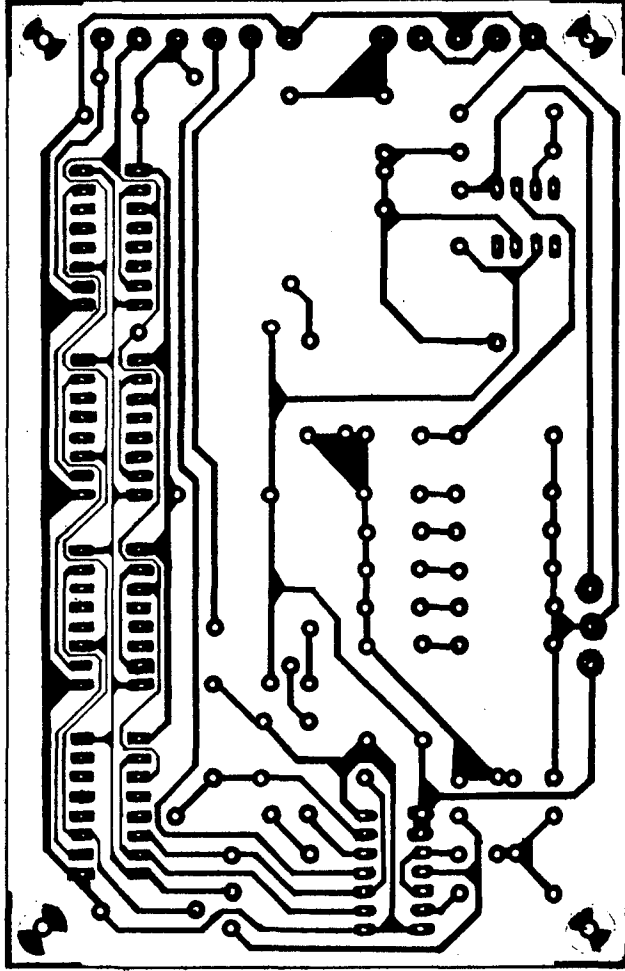
- 1.- PLANO DEL DIAGRAMA DE BLOQUES CON ORDENADOR
- 2.- PLANO DEL GENERADOR DE RUIDO ROSA.

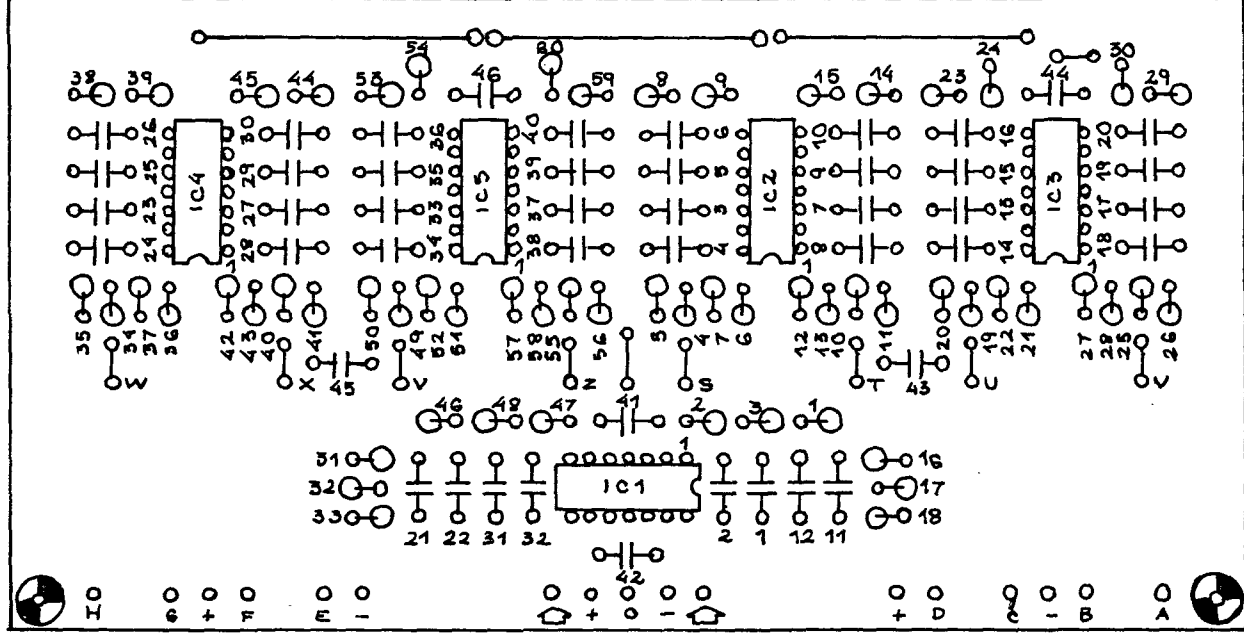


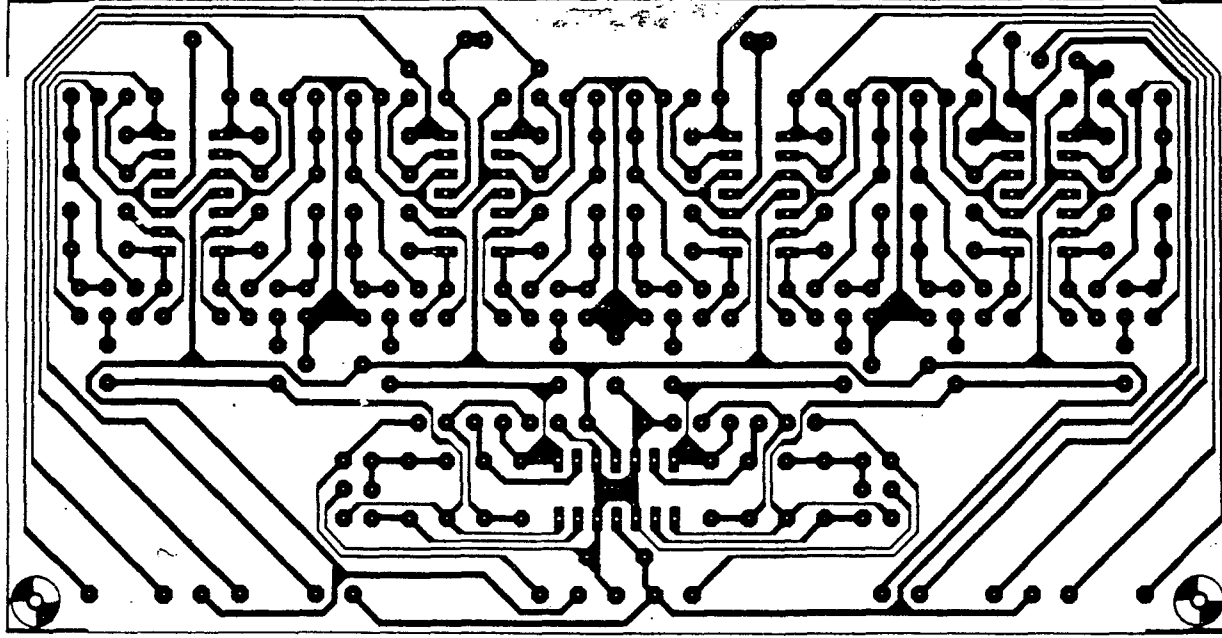


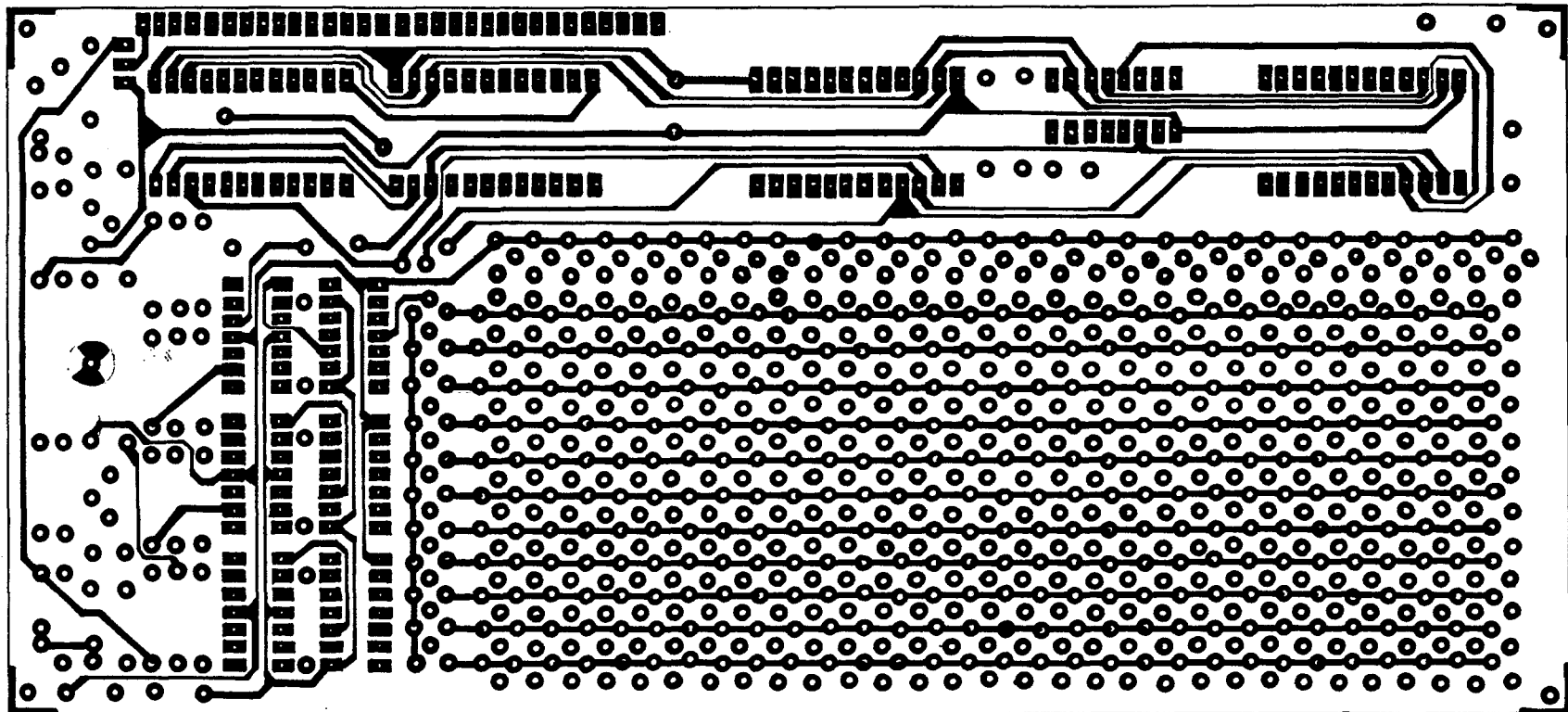


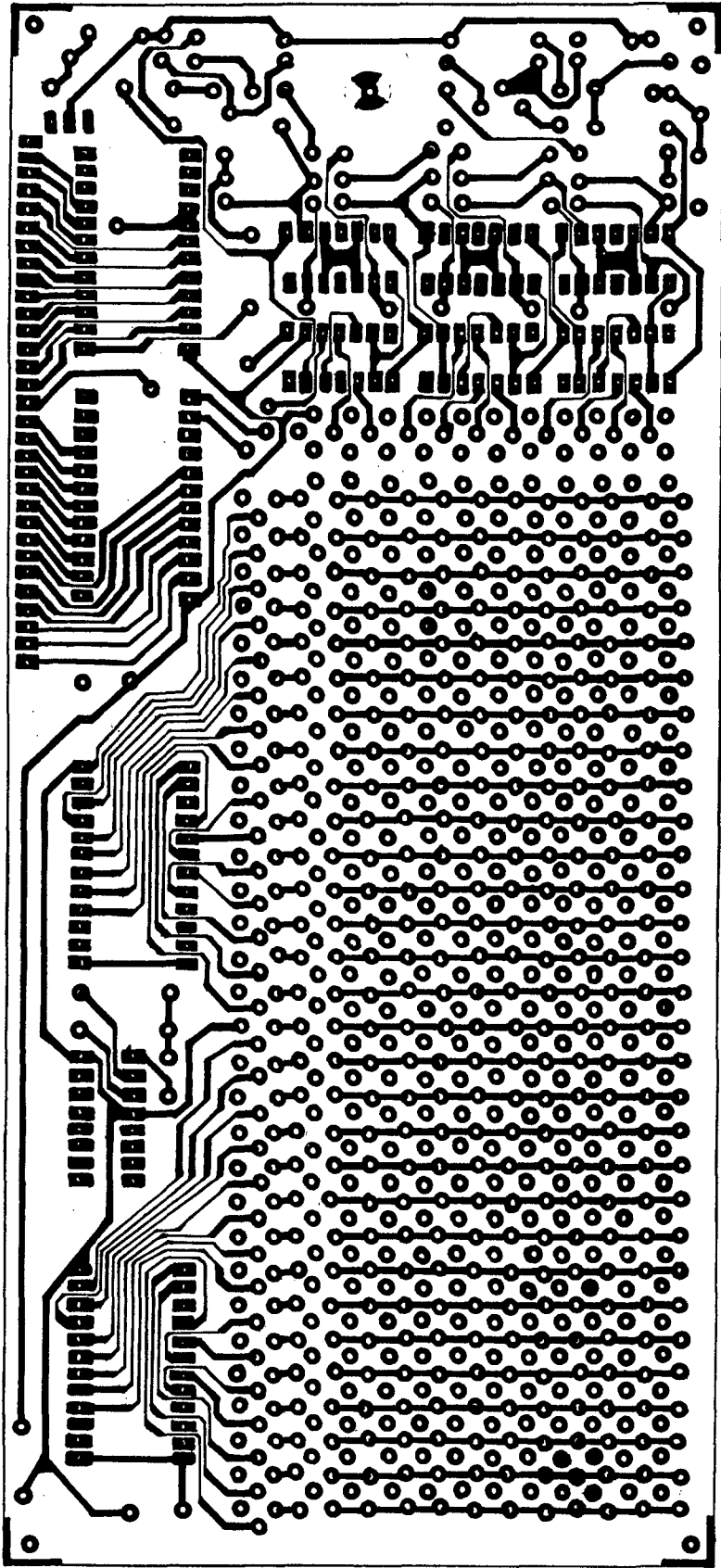


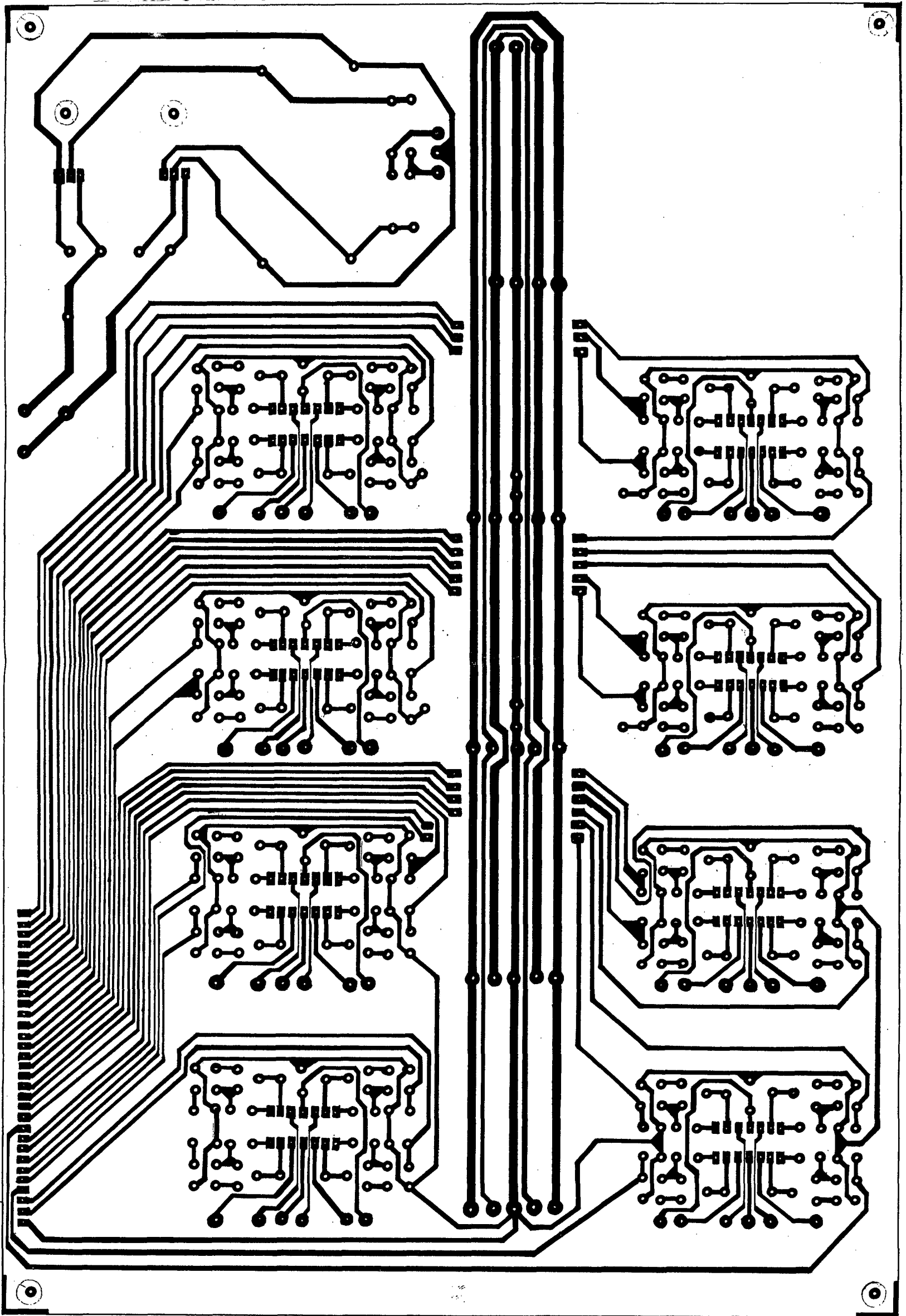




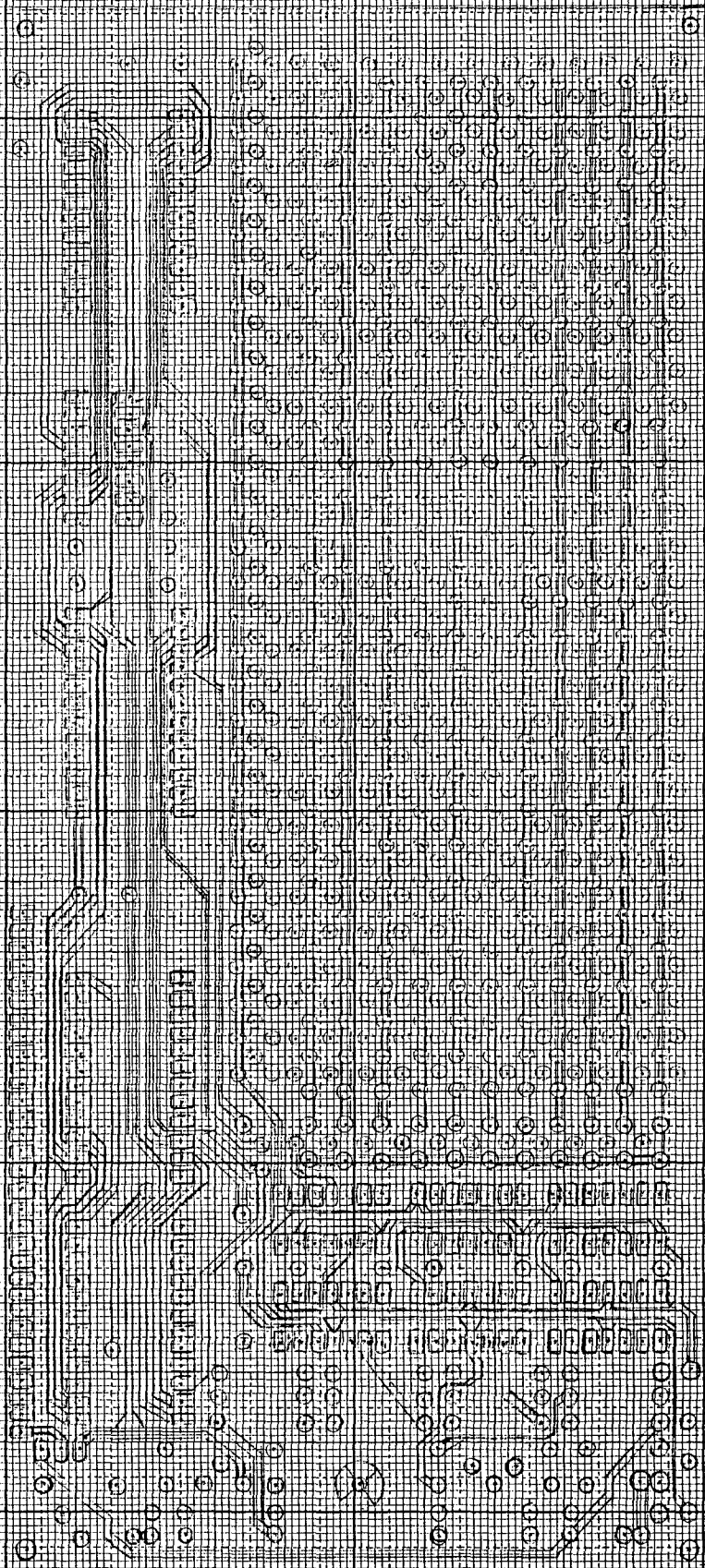


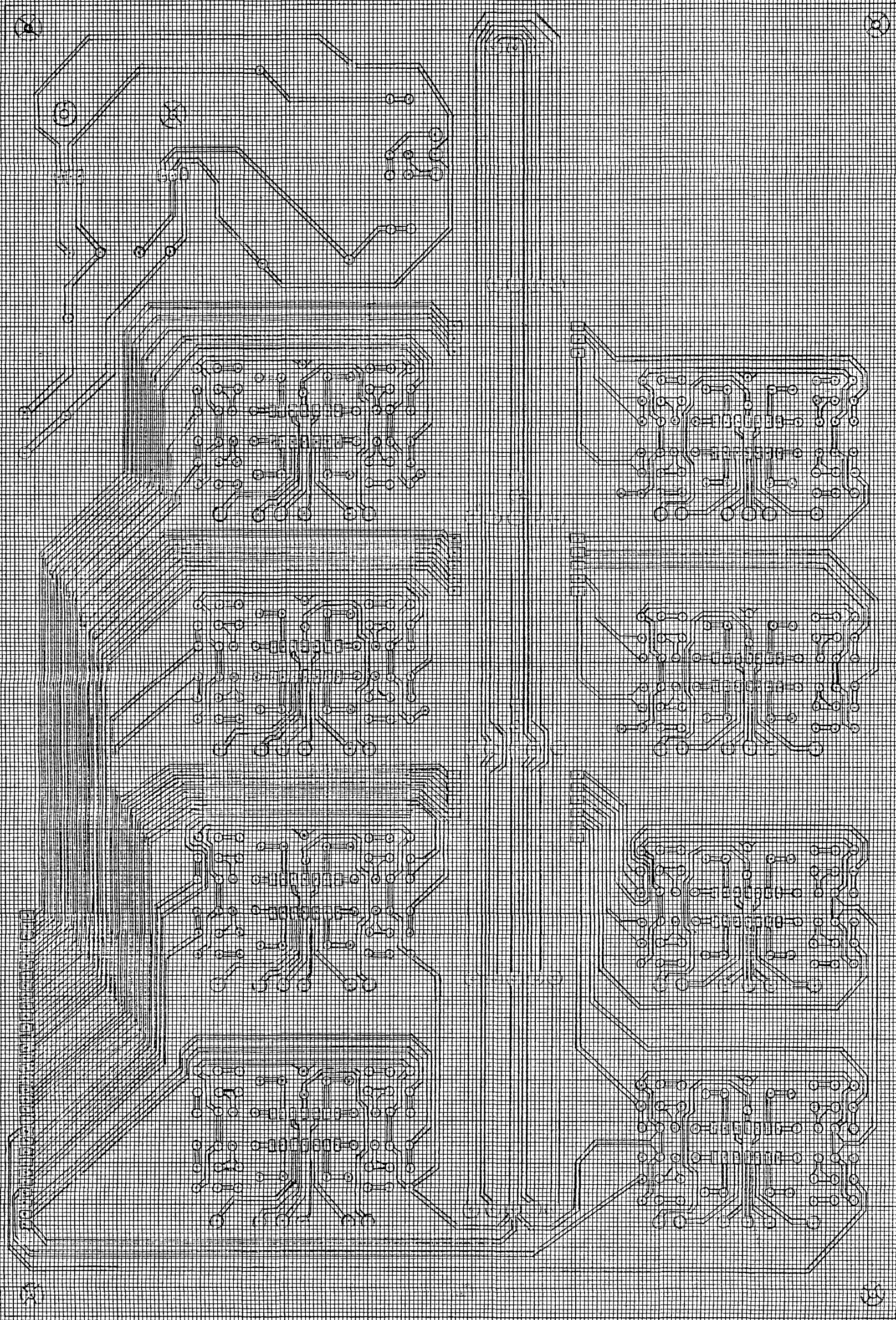










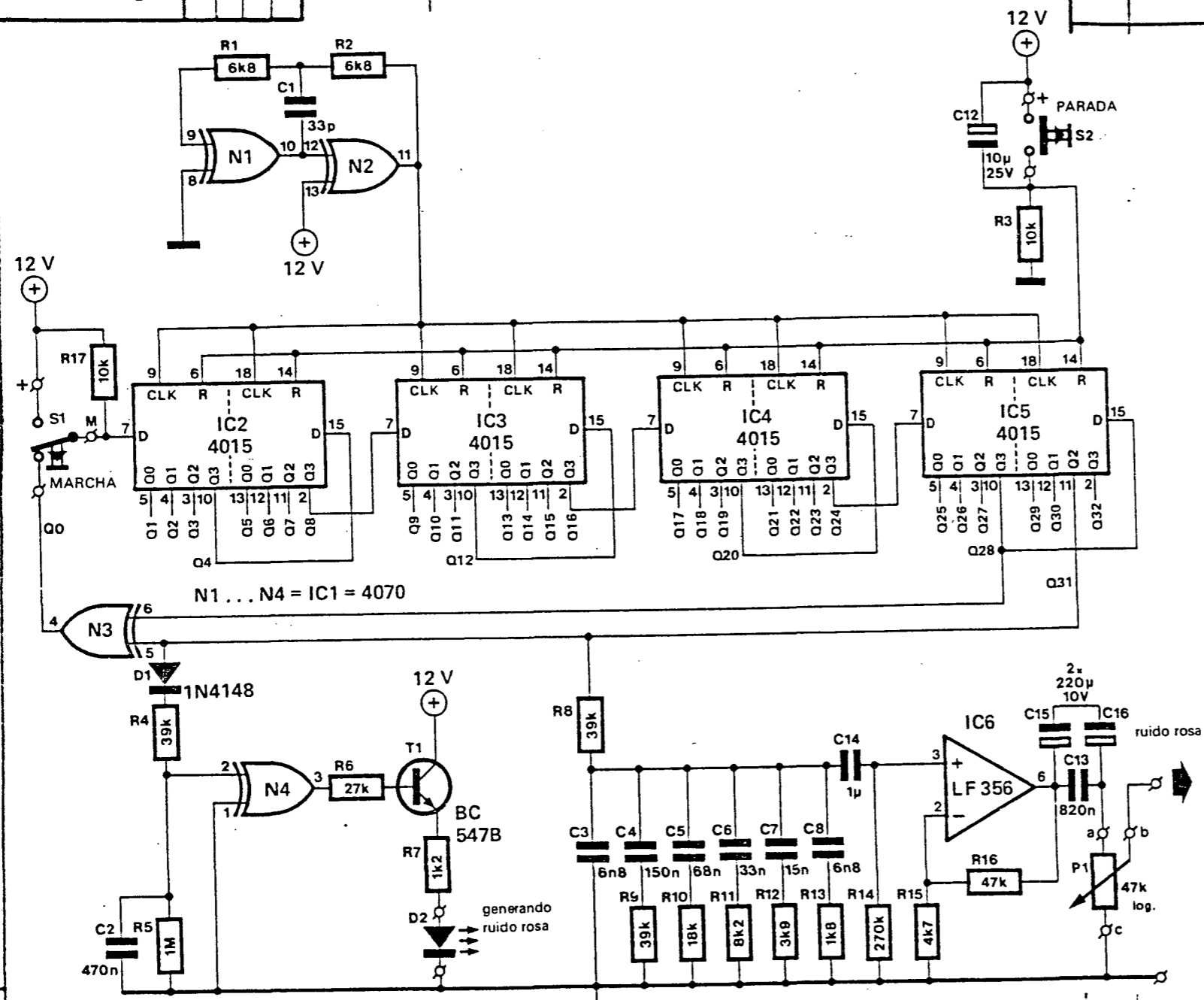


N.

PLANO DE MONTAJE N.º 3

3

NOTAS



N1...N4 = IC1 = 4070

APROBADO

COMPROBADO

DIBUJADO

TOLERANCIAS

REVESTIMIENTO

MATERIALES

PETICIONARIO

FECHA

CANTIDAD

SUPERFICIES

COTAS

ESCALA

denominación

PROYECTO

ANALIZADOR DE AUDIO

N.

N.º	DESCRIPCION	REFERENCIA	MATERIAL	CANTIDAD

Ref	Modificaciones	Fecha	Aprob	Ref	Modificaciones	Fecha	Aprob

160  
100  
50  
40  
30  
20  
10  
0

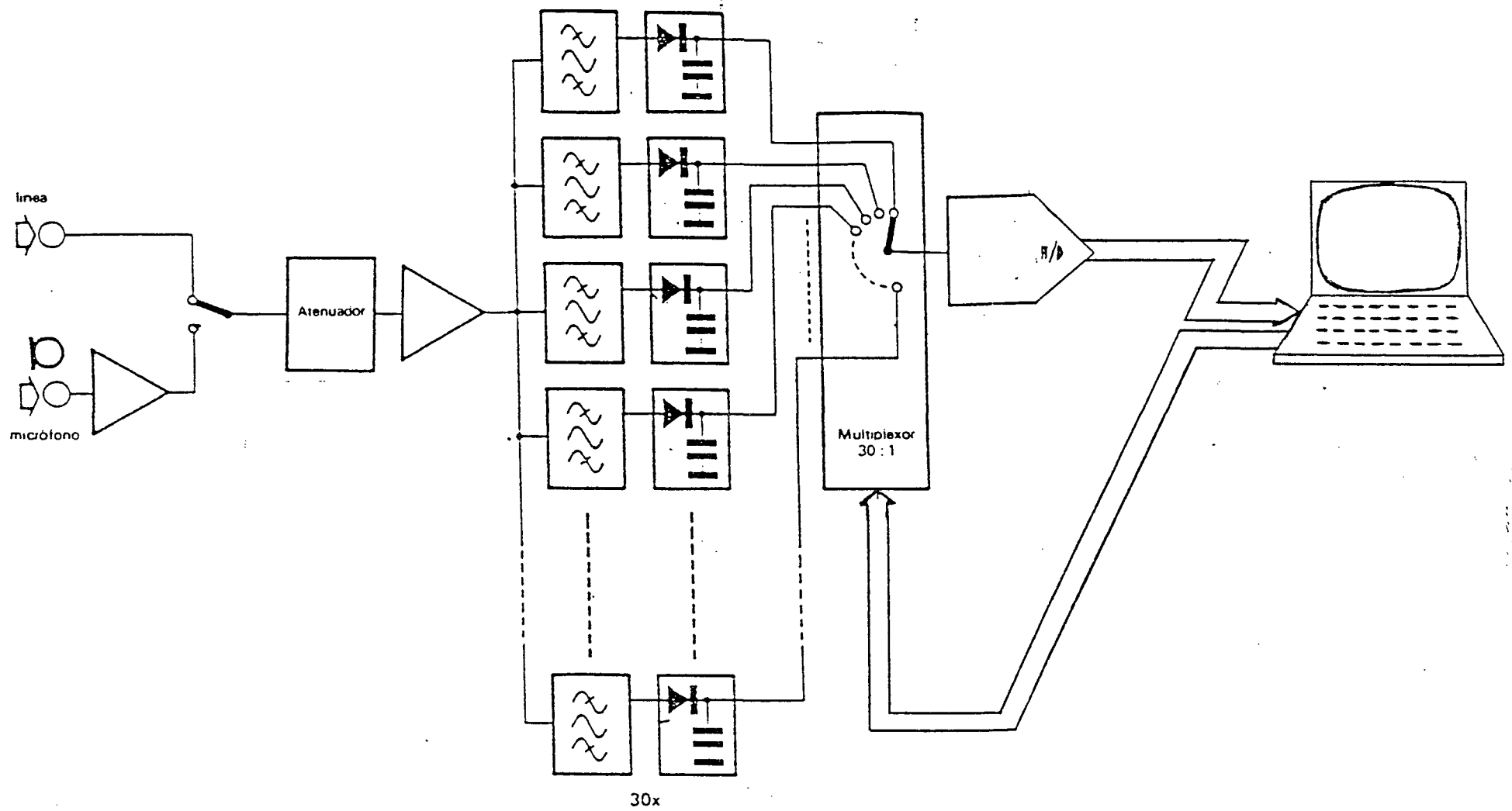
N.

PLANO DE MONTAJE N.º 3




N.º	DESCRIPCION	REFERENCIA	MATERIAL	CANTIDAD

NOTAS



30x

APROBADO

COMPROBADO

DIBUJADO

TOLERANCIAS

REVESTIMIENTO

MATERIALES

PETICIONARIO

PROYECTO

ANALIZADOR DE AUDIO

A3

FECHA

CANTIDAD

SUPERFICIES

COTAS ESCALA

DENOMINACION

N.

Ret	Modificaciones	Fecha	Aprob	Ret	Modificaciones	Fecha	Aprob

