

UNIVERSIDAD POLITECNICA DE CANARIAS
INGENIERIA TECNICA DE TELECOMUNICACIONES
PROYECTO FIN DE CARRERA

TITULO: MANTENIMIENTO DE UN SISTEMA DE CONTROL DE
ESTABILIDAD DE UNA CENTRAL ELECTRICA.
SIMULACION EN TIEMPO REAL

AUTOR:

TUTOR:

Emilio Garcia Bethencourt

Sebastián Suárez Gil

FEBRERO 1988

I N D I C E

	<u>PAGINA</u>
PREFACIO.....	1

PARTE I

ESTUDIO TEORICO

* FIABILIDAD Y MANTENIMIENTO DE UN SISTEMA.....	6
- INTRODUCCION.....	6
- FIABILIDAD.....	7
- FALLO.....	8
- REDUNDANCIA.....	10
- MANTENIMIENTO.....	20
* INSTRUMENTACION PARA MICROPROCESADORES.....	27
- LA SONDA LOGICA.....	28
- GENERADOR DE IMPULSOS LOGICOS.....	28
- RASTREADOR DE CORRIENTE.....	29
- ANALIZADOR DE ESTADOS LOGICOS.....	30
- ANALISIS DE FIRMAS.....	30
- EL ANALIZADOR DE FIRMAS.....	32
- SISTEMA DE DESARROLLO.....	39

* LOCALIZACION DE AVERIAS EN SISTEMAS CON MICROPROCESADOR.....	48
- PROBLEMAS EN LAS REPARACIONES DE SISTEMAS CON MICROPROCESADOR.....	49
- PROBLEMAS ESPECIFICOS DE LOS SISTEMAS CON MICROPROCESADOR.....	51
- PROGRAMAS DE AUTOTEST.....	58
- EL RECONOCIMIENTO DEL PROBLEMA.....	67
- LAS AVERIAS DE CIRCUITOS INTEGRADOS.....	70
- TECNICAS GENERALES DE REPARACION DE AVERIAS...	75

PARTE II

ESTUDIO DE UN SISTEMA DE CONTROL DE ESTABILIDAD

* DESCRIPCION DEL SISTEMA.....	86
- FINALIDAD.....	86
- HARDWARE	92
- SOFTWARE.....	104
- ORGANIGRAMAS.....	121
- PROGRAMAS.....	121
* ESTUDIO DE ALGUNOS CIRCUITOS LSI UTILIZADOS.....	122
- EL MICROPROCESADOR 8085.....	124
- EL INTEGRADO 8155.....	143
- LA USART 8251.....	153
- EL CONVERTIDOR ADC0808.....	169

* INFLUENCIA DE LOS CAMPOS ELECTROMAGNETICOS.....	176
- APANTALLAMIENTO.....	178
- ULTIMOS DETALLES.....	188

PARTE III

MANUAL DE MANTENIMIENTO

SIMULACION EN TIEMPO REAL

* INTRODUCCION.....	190
* TEORIA DE FUNCIONAMIENTO.....	193
* CARACTERISTICAS Y LIMITACIONES DEL DISEÑO.....	201
* PRUEBAS DE MANTENIMIENTO.....	206
- MEDICIONES.....	207
- AUTOTESTEO DEL SISTEMA.....	215
* POSIBLES AMPLIACIONES Y MEJORAS.....	218
* SIMULACION EN TIEMPO REAL.....	224
- ORGANIGRAMA DE SIMULACION.....	240
- PROGRAMAS DE SIMULACION.....	240

APENDICE :

- PLANOS ELECTRICOS
- CARACTERISTICAS DE COMPONENTES

BIBLIOGRAFIA

P R E F A C I O

La empresa eléctrica UNELCO, a través de don J. A. Torres, ha propuesto la idea de un sistema de control de estabilidad destinado a una central eléctrica.

La necesidad de un control así surge como consecuencia de la poca seguridad ofrecida por el método tradicionalmente utilizado para la conservación de la estabilidad en centrales de este tipo: se trata del método de los relés de mínima frecuencia.

Es indudable el enorme desarrollo que los microprocesadores, los microcomputadores y los sistemas basados en ellos ha experimentado en nuestros días, y este es el camino al que sigue tendiendo. El coste se reduce, y sus posibilidades aumentan. La versatilidad, manejabilidad y facilidad de interconexión los hace idóneos para la realización de tal control, presentando enormes ventajas frente al sistema tradicional. Permite, por ejemplo, el

diseño modular, muy cómodo a la hora de implementación, mantenimiento y reparación.

En este proyecto, más que referirnos a la forma en que se ha llevado a cabo la implementación hardware y software del mencionado sistema (aunque también serán tratados estos puntos), se profundiza en el mantenimiento del mismo, así como la forma de llevar a cabo una simulación adecuada de las condiciones reales de funcionamiento.

Con intención de ayudar a tomar una idea general de lo que se expone en este libro, y las razones de ello, se comenta a continuación las partes en que está estructurado.

Se compone de tres partes principales:

* La primera parte hace un estudio teórico de temas relacionados con el mantenimiento de sistemas basados en microprocesadores. Es importante el concepto de fiabilidad, por ser ésta una característica imprescindible de cualquier sistema, tanto más este, debido a la responsabilidad que conlleva.

Se indica un conjunto de técnicas de mantenimiento, de reparación, así como se relacionan instrumentos útiles

para dichas tareas.

* La segunda parte analiza el sistema de control de forma más concreta. Se describe el sistema detalladamente, explicando en primer lugar los objetivos que persigue, y después las implementaciones hardware y software que nos han ayudado a alcanzarlos.

Como ayuda al mantenimiento se ofrece además una explicación de la forma de manejo de algunos circuitos integrados que han sido empleados en el diseño hardware, y que además están muy vinculados al software por su característica de programabilidad.

El porqué de este análisis es muy sencillo: a la hora de llevar a cabo un estudio del mantenimiento del sistema y la forma de realizarlo con éxito es necesario un buen conocimiento previo de cada parte del mismo. Existen ciertos circuitos considerados de cierta complejidad para aquellos que nunca los han manejado; con su estudio pretendemos facilitar el aprendizaje, y al mismo tiempo ayudar a justificar las razones de ciertas formas de implementación software y hardware utilizadas.

No falta la mención de algunos problemas inevitables en la puesta en funcionamiento del sistema, y la forma de

minimizarlos.

* La tercera parte pretende ser un manual de mantenimiento. Además se comentan posibles mejoras y ampliaciones del sistema.

Por último se desarrolla una simulación en tiempo real del dispositivo. Los motivos de ello son muy importantes, como se podrá observar en esta parte del proyecto. La realización de una simulación así se hace imprescindible, por la imposibilidad de verificar el funcionamiento del sistema bajo condiciones reales. Los medios de que se dispone para la comprobación del buen funcionamiento son bastantes mediocres, no pudiéndose forzar todas los posibles casos, o al menos los más importantes, a fin de observar las respuestas del sistema ante ellos.

Esta circunstancia ha obligado a realizar una segunda versión de los programas, la llamada "versión para la simulación", quedando solucionado así el problema mencionado. Además, esta versión nos proporciona una excelente presentación del producto al usuario, ya que nos da la oportunidad de demostrar la efectividad y alcances del sistema.

Se adjuntan también las características de los diversos

elementos que forman parte del hardware, dentro de un apéndice final, así como los planos eléctricos del sistema implementado.

P A R T E I

E S T U D I O T E O R I C O

F I A B I L I D A D Y M A N T E N I M I E N T O

D E U N S I S T E M A

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

Aparte del diseño de un sistema y de su posterior puesta en funcionamiento, es importante pensar en el futuro del mismo. El usuario de un sistema adquiere éste para utilizarlo en cierto propósito, y espera que su producto funcione durante un tiempo razonable, de forma que le sea rentable.

Es por tanto que el fabricante debe ir tras dos objetivos en cuanto al futuro del producto que vende: conseguir la máxima fiabilidad del mismo, y proporcionar información o medios para su mantenimiento. La posibilidad de fallos en el sistema ha de ser minimizada.

FIABILIDAD

Se puede considerar la fiabilidad como la mejor medida cuantitativa de la integridad de un componente, o de un sistema determinado. La fiabilidad da la probabilidad de que los componentes o sistemas cumplan las funciones para las cuales han sido diseñados, sin fallos, en condiciones de funcionamiento determinadas, durante el periodo de tiempo deseado o fijado para ello.

Podríamos dar una definición de fiabilidad como la probabilidad condicionada a un nivel de confianza dado de que un componente o equipo funcione correctamente durante un tiempo determinado, en las condiciones de trabajo para las que fue diseñado. La fiabilidad de un sistema o equipo vedrá determinada por la de sus componentes.

FIABILIDAD Y COSTO

De entre los posibles valores de fiabilidad de un componente o sistema, existe uno cuyo costo es mínimo. Se trata del nivel de fiabilidad óptimo. Con respecto a este valor óptimo se pueden dar los siguientes casos de nivel de fiabilidad:

- fiabilidad inferior a la óptima.
- fiabilidad óptima.
- fiabilidad superior a la óptima.

FALLO

Se llama fallo a la terminación de la capacidad de un dispositivo o elemento para la realización de su función, dentro de unos límites definidos de actuación.

Una clasificación de los fallos podría hacerse desde tres puntos de vista:

1) Según la dependencia hay dos tipos:

- primarios: aquellos que son independientes, es decir, no dependen de otros.
- secundarios: cuando dependen de otros.

2) Según el instante en que se produzcan, los fallos pueden ser:

- precoces o de mortalidad infantil.
- aleatorios.

- de desgaste o por envejecimiento.

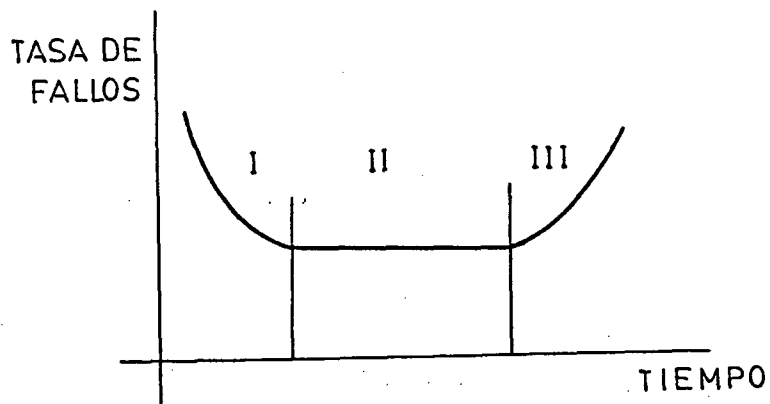
3) Según la forma en que se producen pueden ser:

- catastróficos, súbitos y totales.

- por degradación o deriva.

CURVA DE MORTALIDAD O DE BANERA

En la mayoría de los casos, una curva en forma de bañera es representativa de la tasa de fallos de un componente o sistema, en función de su edad. Se puede distinguir en esta curva tres periodos durante los cuales se producen los distintos tipos de fallos comentados, como se ve en la figura.



El primer periodo corresponde a los fallos infantiles, y se extiende entre O y T. En él la tasa de fallo decrece rápidamente. Aparecen los defectos debidos a errores de fabricación o de concepción, tales como técnicas de fabricación malas, mano de obra mediocre, materiales de calidad inferior a la señalada en las normas, control de calidad mediocre,...

El segundo periodo se denomina vida útil, y en él la tasa de fallo es prácticamente constante. Entre las causas de fallo en este periodo se encuentran: factores de seguridad insuficientes, errores humanos en la utilización, mala aplicación, causas inexplicables (fallos producidos por el azar).

El tercer periodo se denomina de desgaste o envejecimiento, con una tasa de fallo creciente y caracteriza el final de la vida útil del producto. Como causas principales de fallo están: la edad, el desgaste, degradación, fluencia, fatiga,... Al contrario que en el periodo infantil, en éste el producto cada vez es peor.

REDUNDANCIA

Una meta en fabricación es conseguir un producto

resistente a fallos. Se podrían citar dos técnicas principales:

- Diseño de circuitos o sistemas libres de fallos, lo que de momento constituye una utopía.
- Incorporación de redundancia para el supuesto de darse fallo, reemplazando el elemento fallido.

Un sistema o componente de un sistema es redundante cuando contiene varios componentes para cumplir una función dada, pudiendo ser cumplida dicha función por uno de los componentes, es decir, existe una repetición de medios para realizar la misma función.

La redundancia tiene por objeto aumentar la fiabilidad del sistema en el caso de que un componente indispensable tuviese una fiabilidad pequeña. Este método no puede ser aplicado de manera sistemática, por razones técnicas, de peso, volumen, consumo,...

Según la forma de realizar la redundancia podemos ver dos tipos:

- redundancia activa.
- redundancia pasiva o de reserva.

Por redundancia activa se entiende aquellos sistemas en

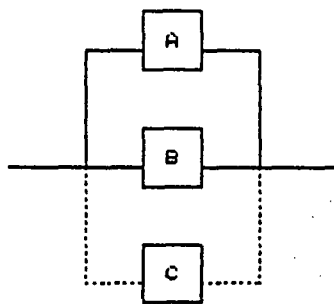
los que todos los elementos trabajan simultáneamente, de forma permanente, pudiendo ser una redundancia activa parcial, total o mayoritaria.

Por redundancia pasiva se entiende aquellos sistemas en los que un elemento está trabajando y el resto está en reserva, esperando a entrar a funcionar si se produce una anomalía del primero. La entrada en funcionamiento se realiza mediante una conmutación.

Existen diferentes técnicas de redundancia:

a) Redundancia simple paralelo.

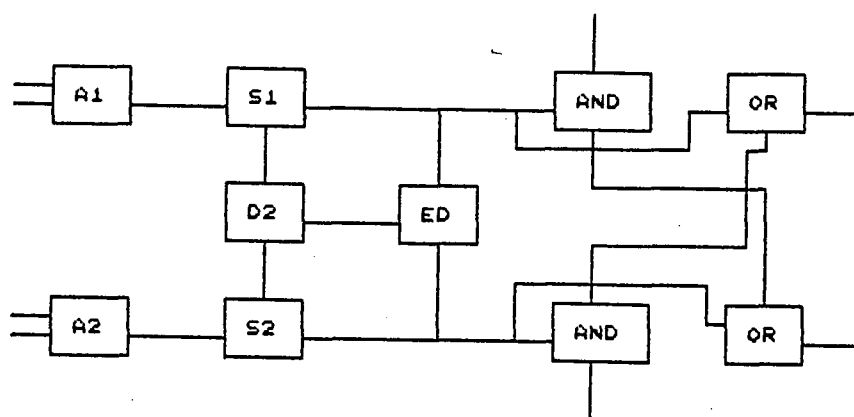
Si falla cualquiera de los elementos, aún quedarán otros caminos idénticos a través de los otros elementos paralelos.



b) Redundancia dúplex.

Las secciones A1 y A2 operan en paralelo y pueden

emplearse en redundancias activas o como elemento separador. Se coloca un conmutador a las salidas de cada elemento, S1 y S2. Los conmutadores son controlados por el elemento DL, que les abre paso hacia las líneas que atienden. Un detector ED se sitúa a la salida para que chequee ambas secciones, y en el supuesto de detectar alguna variación se arranca una rutina para determinar e inhibir el elemento fallido, conmutándose el switch correspondiente. Así se permite la salida por la puerta AND del elemento activo; de esta forma las salidas de los puntos OR de ambos sistemas son las mismas aunque esté funcionando sólo uno de ellos.



c) Redundancia binodal.

Es una técnica que se aplica principalmente a aquellos equipos donde se requiera protección contra

fallos de cortocircuitos o circuitos abiertos principalmente.

Una conexión serie de elementos paralelos redundantes provee de protección contra cortocircuitos y circuitos abiertos. El modo A protege ante fallos de circuito abierto, y el modo B de cortocircuito.

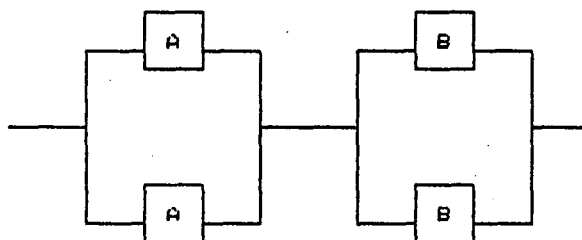


FIG. A

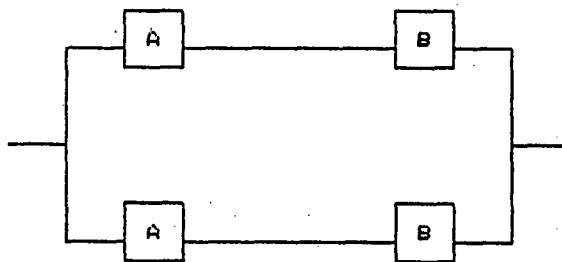
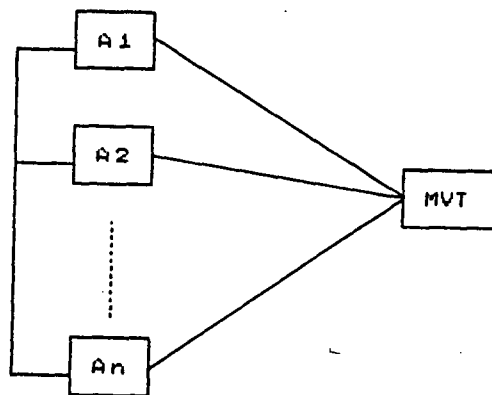


FIG. B

d) Redundancia de votación por mayoría.

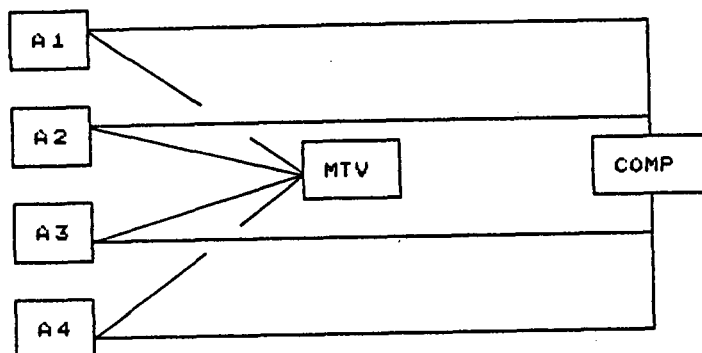
Un problema común en los elementos que emplean redundancia es determinar qué equipos están libres de

fallos y cuáles no. La redundancia de votación por mayoría es una estructura paralela básica con elementos votados por mayoría, que compara la salida de los n elementos y da por salida la que prevalece. Una solución es emplear un número impar de circuitos paralelos y asumir que la mayoría de ellos son correctos, y por lo tanto, en caso de discordancia, determinar mediante la mayoría cuáles están fallando.



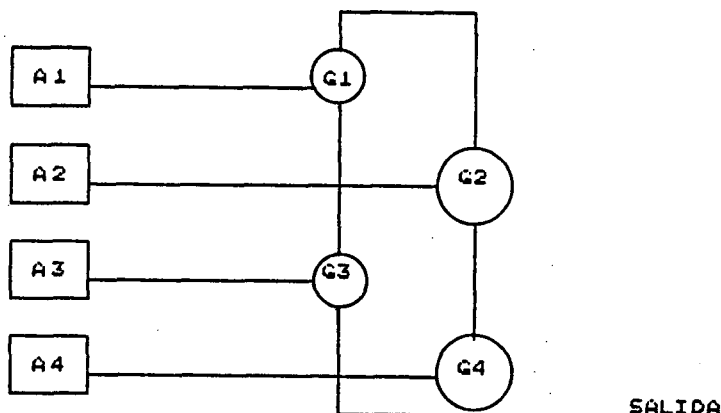
e) Lógica de mayoría adaptativa.

Es un complemento de la función anterior de votación por mayoría mediante una red de comparación y de conmutación, que cuando un elemento no es concordante con el resto, entonces lo inhibe permaneciendo el resto de los equipos en activo.



f) Redundancia de conexión de puerta.

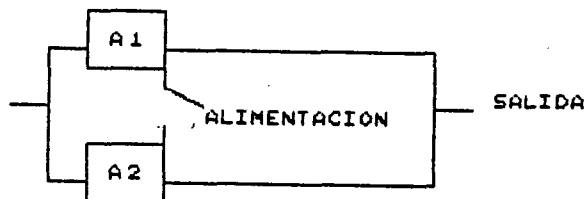
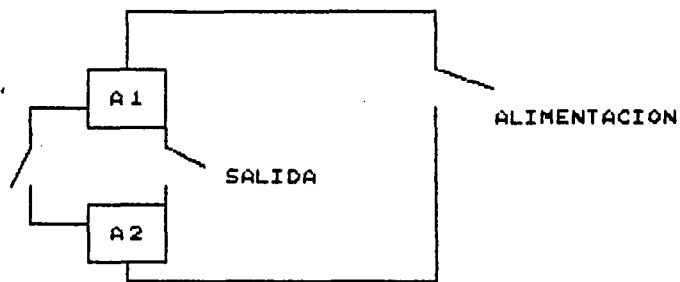
Es similar a la votación por mayoría, pero sólo para elementos binarios. Sus salidas se conectan a la puerta, que es un componente cuyo fallo no afecta al sistema, esto es, si falla la puerta el sistema se comporta como si fallara el elemento a la que está conectada.



c) Redundancia pasiva.

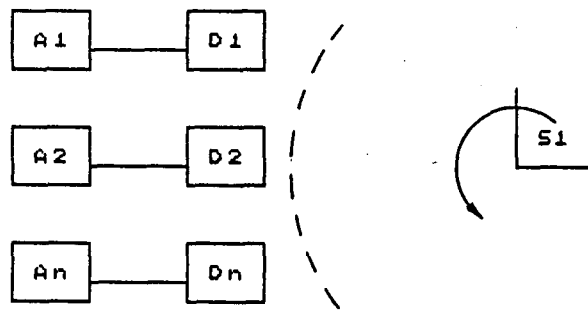
En un determinado momento sólo existe un elemento activo en el circuito y el resto de los elementos están desconectados del mismo en reserva. Existen dos variantes:

- El caso en que un solo elemento está alimentado, y cuando se conmuta al elemento de reserva es necesario conmutar completamente entradas, salidas y alimentación.
- El caso en que todos los elementos redundantes están conectados al circuito, y sólo un elemento redundante activo está alimentado. La conmutación de él al de reserva se hace conmutando sólo la alimentación. Las figuras siguientes ilustran estos dos casos:



h) Redundancia operativa.

Todos los elementos redundantes operan simultáneamente y un sensor detecta las unidades fallidas. Cuando se detecta un error se deja la unidad fallida fuera de servicio.

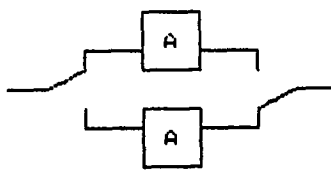


Para sistemas no reparables, una posible configuración viable es la configuración paralela.

En el supuesto de que un equipo sea considerado como un eslabón débil del sistema en el que está integrado, puede conectarse en paralelo con el primero un segundo circuito idéntico, haciendo en este punto que el sistema sea redundante. Estas conexiones se denominan redundancia activa, ya que ambos dispositivos están en funcionamiento. Puede extenderse el caso a n sistemas actuando en paralelo. Si uno de los sistemas dispuestos fallara, entonces un segundo equipo automáticamente se haría cargo

de la función.

Cuando por razones de tipo de interconexión electrónica no se puede aplicar la redundancia activa, aún es posible emplearla de forma pasiva, de modo tal que dado un instante de tiempo, sólo existe un elemento funcionando en el sistema, y cuando ese elemento falla se conmuta a un segundo elemento.



FIABILIDAD DE PARTES ELECTROMECHANICAS

La aparición de fallos en equipos y componentes electrónicos sigue una distribución de Poisson, es decir, que los fallos se producen de modo aleatorio y con una constante respecto al tiempo. La práctica, sin embargo, ha demostrado que no todos los componentes presentan una tasa de fallos constante. Este es el caso de los componentes mecánicos y electromecánicos, e incluso para algunos componentes electrónicos. La distribución que mejor se

adapta al comportamiento de los componentes electromecánicos y mecánicos es la de Weibull.

MANTENIMIENTO

La mayoría de los sistemas están sujetos a acciones de mantenimiento a lo largo de su vida útil. Existen dos tipos de mantenimiento, el denominado preventivo y el llamado correctivo.

El mantenimiento preventivo se ejecuta en intervalos regulares y puede contribuir significativamente a mejorar la fiabilidad y la disponibilidad del equipo. No obstante, hay que hacer consideraciones sobre el costo resultante, que en todo caso debe mantenerse en el mínimo. Se aplica sólo cuando existen fenómenos de desgaste; en este sentido, las normas a seguir para el mantenimiento preventivo se basarán en cálculos estadísticos referentes a la probable vida útil de los diferentes elementos. Además de esto, puede ser también necesario aplicar un mantenimiento preventivo en equipos que muestran señales de funcionamiento anormal.

Las políticas de planificación del mantenimiento preventivo pueden llevarse a efecto en dos modos

diferentes:

- 1) con cambios de planificación ante cada acción correctiva.
- 2) sin cambios de planificación ante cada acción correctiva.

Frente a éste, existe el mantenimiento correctivo, donde la ocurrencia de una acción de mantenimiento es una variable aleatoria. El mantenimiento correctivo está directamente relacionado con el cálculo de probabilidades, y la frecuencia con que es necesario efectuar acciones correctivas, siendo proporcional a la tasa de fallos del sistema.

Mantenibilidad es la característica de diseño e instalación que se expresa como la probabilidad de que al ejecutarse una reparación acorde a los recursos y procedimientos especificados sobre un elemento, se efectúe dentro de un periodo de tiempo predeterminado.

Según esta definición, el tiempo es el factor más importante en la mantenibilidad. El objetivo de la mantenibilidad es minimizar este tiempo. El tiempo necesario para el mantenimiento de un sistema se puede dividir en varios apartados:

- A) Tiempo para diagnosticar el fallo.
- B) Tiempo logístico para disponer de herramientas, equipos de medida, personal técnico,...
- C) Tiempo para acceder al elemento fallido.
- D) Tiempo para reemplazar el módulo fallido o repararlo.
- E) Tiempo para inspeccionar, verificar, restaurar las condiciones de trabajo, recargar,...

La mantenibilidad de un equipo se representa por la probabilidad de completar una acción de mantenimiento en un tiempo dado.

El costo del mantenimiento de un equipo y su fiabilidad son en gran parte dependientes de las políticas de mantenimiento.

Si el costo de una acción preventiva es cercano o mayor que una correctiva, no puede emplearse este mantenimiento. Cuando la tasa de fallos es constante, el mantenimiento preventivo no tiene efecto, a no ser que haya redundancia.

A, medida que los sistemas aumentan en coste y

complejidad, ya no es posible tomar en consideración sólo su fiabilidad. Es seguro que la utilización de mejores componentes o la aplicación de redundancia se vean rápidamente limitadas en los sistemas complicados, y será por tanto necesario proceder a su mantenimiento, sobre todo si se prevee una vida larga de los mismos.

A la hora de trabajar con desarrollo de equipos es necesario una prevención de defectos, cuya guía es la siguiente:

- Revisión de especificaciones para comprobar la coherencia entre ellas.
- Determinación y planificación de la fiabilidad del producto determinado: defectuosidad de los componentes que constituyen el producto, posibles redundancias que aumentarían la fiabilidad, determinación de la defectuosidad de los procesos involucrados, efectividad de las pruebas realizadas.
- Calificación de los componentes y productos que intervengan en el sistema a diseñar.
- Control de cambios y aceptación de revisiones.

Los equipos tendrán sus requisitos de suministro eléctrico:

- Requisitos de potencia: se detallarán los requisitos de tensión, frecuencia y potencia para el suministro en c.a., y de tensión y potencia para suministros de c.c., así como cuantos datos se requieran para el suministro de ambas tensiones y fuentes de suministro.
- Requisitos de protección: protección para corriente, para sobretensiones, limitaciones de temperatura.
- Conexionado a tierra, determinándose las formas de: conexión a tierra del producto completo, conexión a tierra del chasis.
- Seguridad del producto: se relacionarán las normas nacionales y/o legislación vigente, y cuando se estime necesario se detallarán situaciones o puntos especiales que pueden afectar a la seguridad de las personas que manejen el equipo.
- Requisitos funcionales: se especificarán las características de funcionamiento del producto, incluyendo un listado de todos los parámetros

relacionados con el funcionamiento, así como sus tolerancias o límites de actuación. Deberá además especificarse las pruebas funcionales a que deba ser sometido el producto y formas de realizarlas, tanto las iniciales como las periódicas.

Deberá incluirse una definición clara de lo que se considera fallo, o la incapacidad del producto para realizar la función o cualquier eventualidad que requiera una acción de mantenimiento correctivo.

Se citarán los tiempos límites requeridos para el mantenimiento preventivo, tales como:

- intervalo mínimo de mantenimiento.
- intervalo mínimo de tiempo programado entre dos intervenciones de mantenimiento preventivo.
- tiempo medio programado para mantenimiento, referido a un periodo de tiempo previsto.

También se debe especificar los límites de tiempo para el mantenimiento correctivo, así como los repuestos y la cantidad necesaria para la disponibilidad durante el periodo de comportamiento correcto (o vida útil) del

producto.

En la práctica se realizan pruebas aceleradas, en las cuales el equipo actúa ante condiciones extremas de funcionamiento. Además se lleva a cabo un control de: soldaduras, cortocircuitos, circuitos abiertos, componentes defectuosos eléctricamente,...

I N S T R U M E N T A C I O N P A R A

M I C R O P R O C E S A D O R E S

Los sistemas basados en microprocesadores, como sistemas digitales que son, necesitan una instrumentación adecuada a sus características, capaces de detectar magnitudes digitales.

Siguen siendo útiles instrumentos de aplicación general, como el t ster y el osciloscopio, pero se hace necesario la disposici n de otros elementos capaces de detectar los distintos estados l gicos que pueden transcurrir a las altas frecuencias de trabajo. Los instrumentos m s comunes o representativos son:

- la sonda l gica
- el generador de impulsos l gicos
- el rastreador de corriente
- el analizador de estados l gicos
- el analizador de firmas

Pero los sistemas basados en microprocesadores tienen otra característica muy peculiar que también ha de ser examinada, tratada y corregida si posee algún error. Se trata de la parte software, que no puede ser analizada con ninguno de los aparatos anteriores. Cómo hacerlo entonces?. Gracias a los sistemas de desarrollo, que permiten la simulación completa del hardware y software de un sistema que posee un determinado microprocesador.

Veamos las características de todos estos instrumentos citados.

LA SONDA LOGICA

La sonda lógica nos indica el estado lógico de un punto del circuito de prueba. Mediante unas lamparillas indicadoras podemos ver cuál es este estado, permitiendo incluso apreciar impulsos breves.

GENERADOR DE IMPULSOS LOGICOS

Como su nombre indica, este instrumento genera impulsos lógicos para ser aplicados a ciertos puntos del circuito

en examen, y así poder ver los resultados producidos por ellos. Estos impulsos se pueden generar de uno en uno o en paquetes.

El generador de impulsos y la sonda lógica forman un buen equipo de trabajo: uno inyecta pulsos y otro detecta los estados producidos como consecuencia de ellos.

RASTREADOR DE CORRIENTE

Es un instrumento que detecta la actividad de la corriente en los nudos lógicos, a través de un captador inductivo instalado en su punta. Ajustando el control de sensibilidad y observando la intensidad luminosa de su lámpara puede identificarse el camino de la corriente y su magnitud relativa, pudiendo localizar así elementos defectuosos, cortocircuitos, fallos en las líneas de buses,...

Los tres elementos comentados tienen una forma física muy similar, en forma de bolígrafo, resultando bastante cómoda su utilización.

ANALIZADORES DE ESTADO LOGICO

Estos instrumentos nos permiten observar, a través de su pantalla o en la de un osciloscopio acoplado a ellos, los diferentes estados lógicos aparecidos en el sistema durante su funcionamiento, y en tiempo real, pudiéndose disparar en cualquier palabra o paso del programa en ejecución (según lo indique el usuario), mostrando los estados lógicos en los alrededores o después de la palabra de disparo.

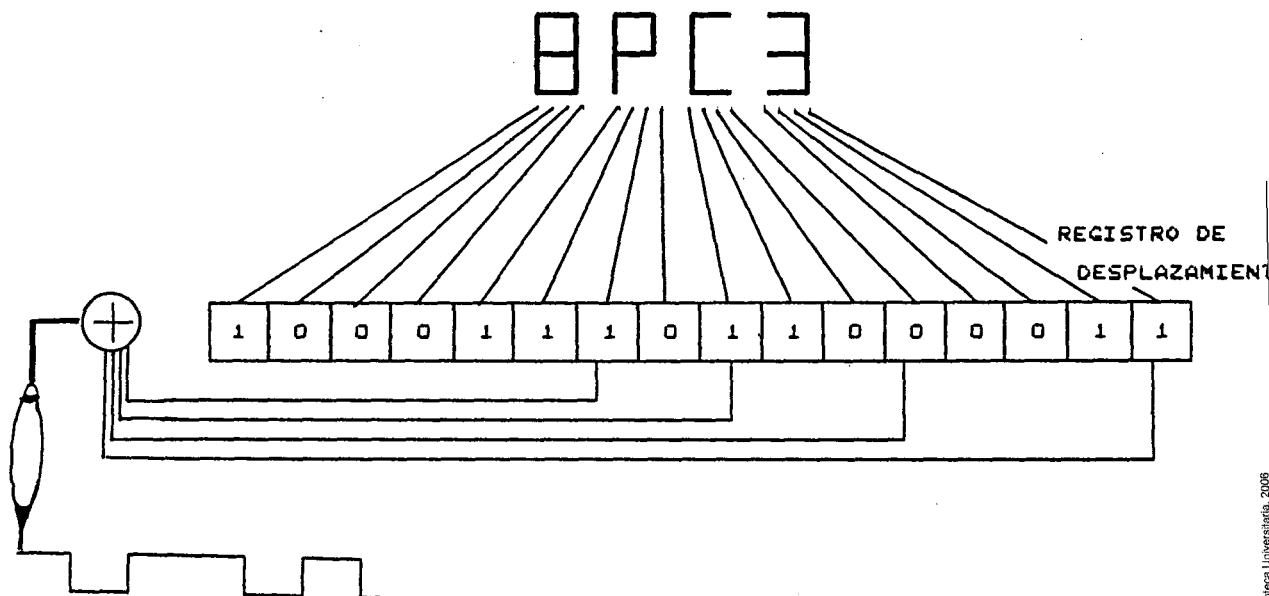
Una vez conectado el analizador al sistema que se prueba, el acceso al bus del microprocesador es muy sencillo; el analizador nos mostrará paso a paso, instrucción a instrucción, los estados generados por el programa.

ANALISIS DE FIRMAS

El mantenimiento a nivel de componente se puede llevar a cabo a través del análisis de firmas. Los circuitos integrados son cada vez más complejos, con tecnologías LSI y VLSI, y su reparación plantea un problema de tanta complejidad como el circuito.

El método de reparación más cómodo y rápido es el sustitución de tarjeta; para ello el sistema se configura en varios módulos o tarjetas reemplazables, y no requiere personal muy cualificado. Pero surge un inconveniente, y es su elevado coste unido a la dificultad de detectar ciertos problemas mediante el intercambio de tarjetas. Una solución más económica y avanzada es el análisis de firmas: se basa en el mismo concepto que el análisis de señales empleado en los circuitos analógicos. En éstos se comparan las formas de onda y tensiones obtenidas en el equipo probado con las anotadas en el esquema, determinando así la avería. El manual de mantenimiento debe dar este tipo de información en sus esquemas para poder determinar el componente que falla.

El instrumento que realiza el análisis de firmas está basado en la técnica de compresión de datos, la cual reduce el tamaño de grupos complejos de datos de 16 bits, transformándolos en un dato de 4 dígitos hexadecimales. Las firmas correctas de un circuito particular se determinan previamente de otro circuito que funcione bien.



EL ANALIZADOR DE FIRMAS

Expliquemos en este apartado un poco la filosofía de un analizador de firmas. Este aparato necesita en principio una serie de estímulos generados por el dispositivo que se prueba o por un elemento externo. La forma de generar estos estímulos internamente es mediante un programa grabado en memoria ROM durante el diseño, cuya finalidad es poder llevar a cabo un análisis de firmas; otra forma de generar estímulos internamente es haciendo trabajar al sistema en funcionamiento libre, o sea, en un lazo

repetitivo, usando un mínimo de elementos y estimulando el máximo número de nudos en el circuito.

Se necesitan tres señales para obtener la firma en un nudo:

- señal de inicio, para comenzar el intervalo de tiempo de medida.
- señal de reloj, que sincronice la toma de datos del analizador con las señales de cada nudo.
- señal de parada: para acabar el intervalo de tiempo de medida.

Las señales de inicio y parada, que limitan el intervalo de tiempo, se pueden obtener de las líneas de direccionamiento, puertas I/O u otras señales. Se eligen unas señales que sirvan para la mayoría de las firmas. El técnico simplemente tiene que ir tomando firmas de nudo en nudo, hasta localizar el componente que tiene la firma de entrada correcta y la de salida incorrecta.

La técnica que se utiliza es la de redundancia cíclica (CRC) empleada en telecomunicaciones para la detección de errores. La firma se obtiene mediante un registro de desplazamiento de n bits con realimentaciones múltiples que se suman con la entrada de datos; el registro de desplazamiento se inicializa con una palabra particular y

se desplaza usando una secuencia de bits específica, obteniendo una palabra residual que siempre es la misma. Un registro de desplazamiento de 16 bits se presenta como 4 dígitos hexadecimales. Obtener una firma distinta a la indicada en el manual no proporciona información sobre la naturaleza exacta del fallo, pero al menos sabremos que éste existe.

Diseño con un análisis de firmas

Las máximas ventajas del análisis de firmas se obtienen cuando se introduce esta técnica en la fase de diseño.

Durante el diseño se reserva una zona de memoria ROM (sobre un 5%) para contener el programa empleado en el análisis de firmas; además son necesarios algunos conmutadores para poner al sistema en modo apropiado para realizar el análisis, resistencias para forzar una instrucción de No Operación (NOP) cuando se pone fuera del circuito la ROM y puentes que puedan quitarse, para aislar partes del circuito.

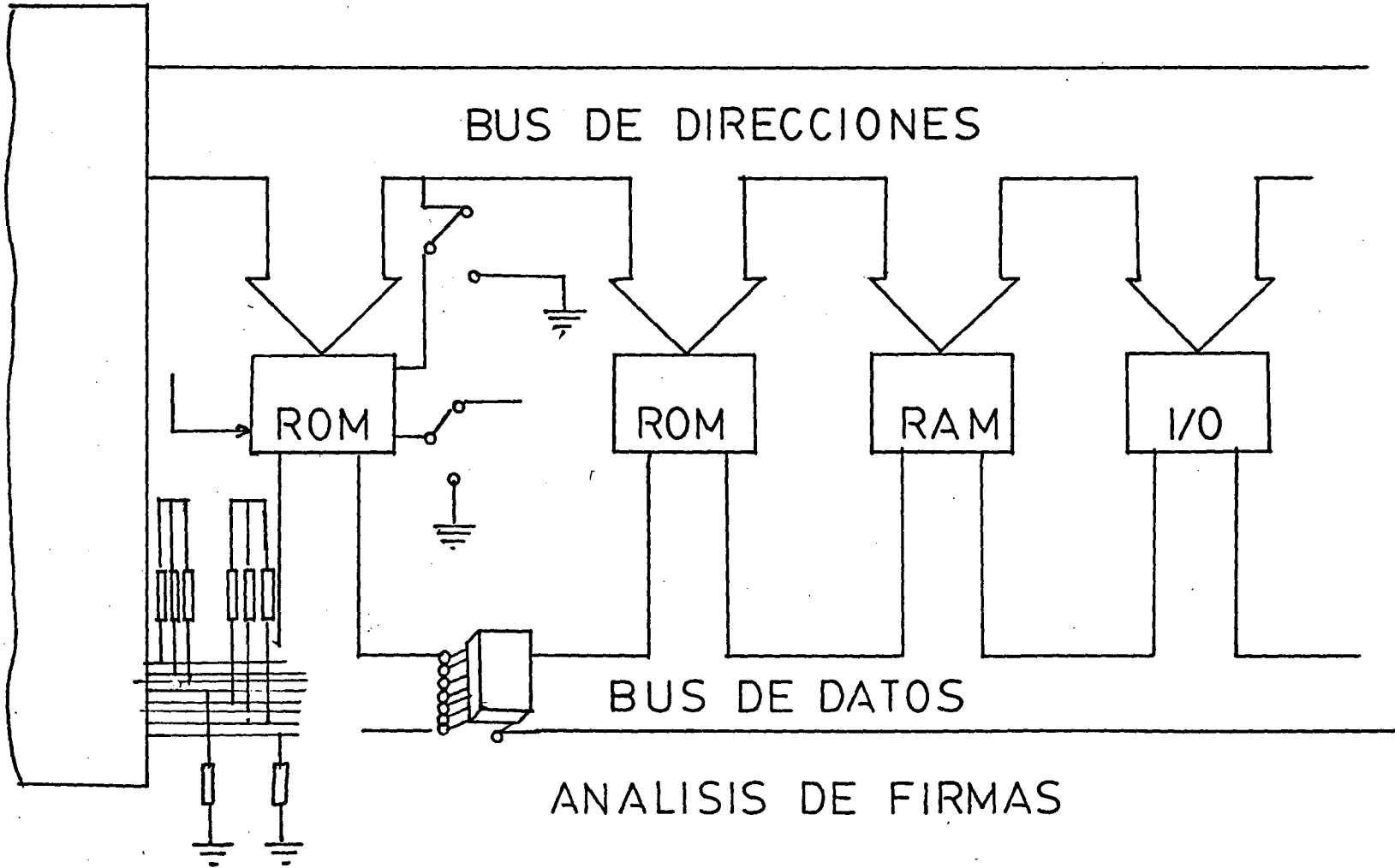
Las firmas correctas que aparecen en el manual de mantenimiento se obtienen tomando varias partes del circuito, que se sabe funcionan correctamente, y anotando los resultados en el esquema del circuito.

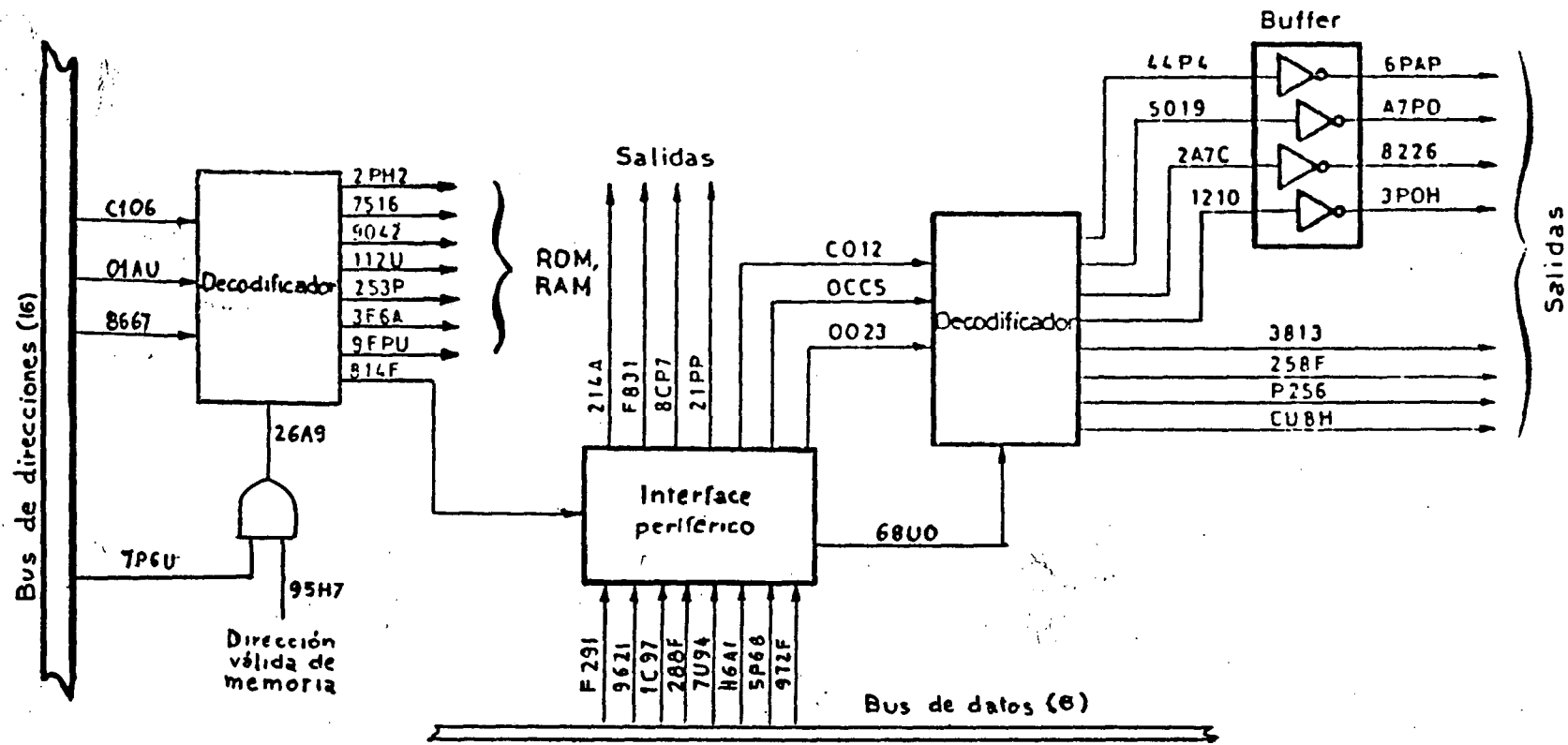
Es necesario romper los lazos de realimentación usando conmutadores, conectores o puentes, para prevenir fallos originados por la realimentación. En la figura de la página 36 aparece un esquema de preparación de un sistema con microprocesador para el análisis de firmas, según lo explicado.

En el caso de que sea el propio microprocesador o la memoria ROM necesaria para el análisis de firmas las causas del fallo, y por tanto no funcione el programa de estímulo, el microprocesador puede generar el estímulo, abriendo el bus de instrucciones/datos y forzando en él una instrucción de No Operación (NOP)

El programa de estímulo comprueba todas las líneas de direccionamiento. Todos los nudos están caracterizados mediante firmas.

La figura de la página 37 muestra un esquema digital con indicación de firmas en los nudos. Se trata de un ejemplo que ilustra lo dicho. Este esquema presenta una representación hexadecimal no estándar (0 1 2 3 4 5 6 7 8 9 A C F H P U), para mayor facilidad de lectura y compatibilidad con displays de siete segmentos, donde podría confundirse el 6 con la B,...





ESQUEMA DIGITAL CON INDICACIÓN DE FIRMAS EN LOS NUDOS

Implementación del análisis de firmas

Aunque una tarjeta no venga diseñada con análisis de firmas, éste se puede implementar fácilmente. Por lo general, el microprocesador está en una tarjeta y la memoria ROM y RAM en otra. En tal caso el bus de entrada de datos al microprocesador está desconectado. La tarjeta del microprocesador puede ser comprobada en funcionamiento libre mientras se toman las firmas del bus de direccionamiento y del resto de los dispositivos de la tarjeta. Esto si la configuración de las tarjetas es modular. Si no es así, para desconectar el bus de entrada de datos del procesador se pueden añadir puentes, zócalos, conmutadores o buffers triestado.

Si la tarjeta no dispone de espacio para esto, se pueden añadir puentes en las salidas del decodificador de direcciones, de forma que cuando se desconecten estos puentes, todos los dispositivos conectados al bus se pongan es estado de alta impedancia. Este método limita la comprobación en funcionamiento libre sólo al contador de direcciones del procesador, dejando sin probar el decodificador y la ROM.

El programa para el análisis de firmas se puede grabar

en la ROM restante del sistema; si esto no es posible se añade otra ROM a la tarjeta, o se cambia la ROM que posee por otra que contenga el análisis de firmas.

El análisis de firmas es un método bastante efectivo para la detección de errores en circuitos digitales. El chequeo lo efectúa a una velocidad del orden de MHz, pues se sincroniza el analizador de firmas con el sistema bajo prueba. Es ideal para localizar fallos relacionados con el tiempo, difíciles de encontrar por otros medios.

SISTEMA DE DESARROLLO

El sistema de desarrollo es un instrumento imprescindible para la puesta a punto de un sistema basado en microprocesador. Nos ayuda a realizar la implementación hardware y software del sistema, pues el funcionamiento de la circuitería LSI y VLSI que forma parte de estos sistemas resulta bastante complejo de examinar.

Los sistemas de desarrollo van desde microcomputadores configurados a nivel mínimo en una placa de circuito impreso hasta sistemas multimicrocomputador sofisticados. Los sistemas de desarrollo de microcomputadores pueden ser de varias categorías:

- sistemas mínimos con microprocesador en una única placa
- sistemas de desarrollo de microprocesadores unipuesto
- sistemas de desarrollo de microprocesadores multipuesto

Los sistemas mínimos con microprocesador en una placa son sistemas de bajo coste; sólo permiten la evaluación de un microcomputador en particular y el desarrollo de programas en lenguaje máquina. Están en una placa de circuito impreso, como se ha dicho, y proporcionan el hardware mínimo. Dentro de esta categoría existen sistemas con cierta complejidad, que incluyen un teclado y display, además de un programa monitor en ROM que controla el sistema a través del teclado. Este teclado puede ser hexadecimal u octal, y mediante él cargamos el programa a ejecutar en posiciones de memoria RAM. El programa monitor hace posible la modificación de los registros internos y los contenidos de memoria, así como la ejecución del programa en desarrollo desde cualquier posición. Se pueden establecer puntos de ruptura que transfieran el control al monitor.

En los sistemas más complejos se dispone incluso de una

interface para teletipo. Pero no disponen de un ensamblador, aceptando sólo código máquina; poseen poca capacidad de memoria y son prácticos solamente a la hora de familiarizarse con un microprocesador particular o para el desarrollo de pequeños programas de aplicación. Un ejemplo de este tipo de sistema es la placa SDK-85 de Intel, para el desarrollo del microprocesador 8085.

Los sistemas de desarrollo unipuesto tienen el aspecto de un minicomputador. Poseen consola de entrada de datos (teclado alfanumérico) y de salida de datos (terminal CRT o teletipo); permite la expansión de las posibilidades del sistema a través de tarjetas adicionales conectables a él. En ROM posee un programa monitor, y tiene la suficiente memoria RAM para soportar un editor y un ensamblador. El almacenamiento de datos se puede realizar en diskettes; en ellos se guardan el editor, el ensamblador, compilador, sistema operativo, ficheros de programas,... La mayoría de los sistemas de desarrollo de microprocesadores cuentan con un programador de EPROM, que posibilita grabar el programa realizado para su ejecución en un sistema microprocesador.

Los sistemas de desarrollo más sofisticados utilizan múltiples microprocesadores, que comparten un bus común y unos dispositivos de E/S. Uno de los microprocesadores, el

maestro, maneja todos los servicios del sistema: manejo de ficheros, editor de textos, E/S del sistema, utilidades del sistema y funciones de depuración. Los otros microprocesadores son esclavos y tienen asignadas ciertas tareas más específicas. A continuación se explica con más detalle las fases de programación en un sistema de desarrollo.

Fases de programación en un sistema de desarrollo

La confección de un programa consta de una serie de fases que no siempre son las mismas, pues dependen de los instrumentos de ayuda de que se disponga, de qué se trate de resolver y del lenguaje de programación empleado. En general se siguen los pasos descritos a continuación, hasta que el programa queda preparado.

1. Definición del proyecto.-

En primer lugar se ha de conocer con detalle las características del proyecto a realizar, qué debe ejecutar y qué dispositivos debe gobernar. Si esta fase no está bien definida, difícilmente se podrán elaborar los programas necesarios.

2. Diagrama de flujo u organigrama.-

Se realiza una secuencia de las operaciones elementales que habrán de ejecutarse mediante las instrucciones. Esta secuencia se representa gráficamente mediante una simbología establecida.

3. Conversión del organigrama en instrucciones.-

Las operaciones elementales expuestas en el organigrama se pasan a instrucciones disponibles en el lenguaje que se vaya a utilizar. El lenguaje a emplear puede ser ensamblador o de alto nivel. El primero ofrece las ventajas de estar más optimizado y de poder minimizar el tiempo de ejecución, pero para programas muy complicados resulta bastante engorroso utilizarlo. El lenguaje de alto nivel es el normalmente utilizado, pues es más adecuado para realizar los programas por su comodidad y mayor claridad.

4. Ensamblaje o compilación.-

Dependiendo de si hemos elaborado el programa en lenguaje ensamblador o en lenguaje de alto nivel, el paso siguiente será hacer un ensamblado o compilado del mismo, respectivamente. Esto consiste en pasar el programa a

código objeto directamente ejecutable por el microprocesador, generando un programa que ya puede correr.

5. Linkado.-

Si hemos realizado un programa con varios módulos, éstos se deben entrelazar para tener una unidad ejecutable. El proceso de enlace lo realiza el programa LINK del sistema operativo. Toma todos los módulos objeto y los transforma en un solo programa objeto.

6. Depuración.-

Una vez confeccionado el programa se hace correr sobre el sistema de desarrollo para verificarlo. El debugger o depurador nos permite rectificar o mejorar partes del programa hasta dejarlo funcionando de forma deseada. Con este programa de utilidad podemos acceder a posiciones de memoria para examinar su contenido, o para cambiarlo, podemos correr el programa hasta donde se desee, estableciendo puntos de parada,...

Si el programa ha sido depurado totalmente, un programador de EPROM permite grabar el código objeto en una memoria EPROM.

7. Emulación y ejecución.-

Los microprocesadores múltiples en un sistema de desarrollo hacen posible la emulación en circuito; el emulador es otro programa de utilidad que suministra el sistema de desarrollo y permite el depurado total de un sistema microprocesador que se esté diseñando, ayudando a la implementación conjunta de software-hardware.

El prototipo hardware diseñado se utiliza junto con el sistema de desarrollo. Se retira el microprocesador del prototipo y se inserta en su lugar un conector enlazado al sistema de desarrollo mediante un cable; además de reemplazar el microprocesador del prototipo, esta conexión hace de interface entre el bus del sistema de desarrollo y el del prototipo. Un microprocesador esclavo funciona como el microprocesador del prototipo.

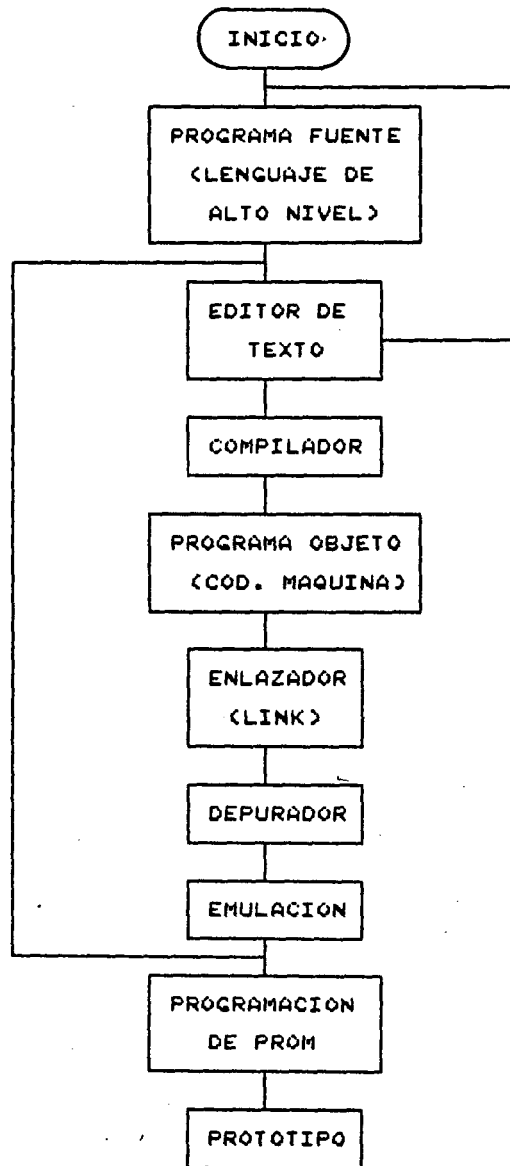
La emulación en circuito se puede emplear desde fases previas, depurando conjuntamente hardware y software, en lugar de un desarrollo por separado de ambas partes y una integración final.

En los estados de desarrollo iniciales el sistema de desarrollo emula el circuito y no es necesario el hardware del prototipo; poco a poco se van completando porciones

del prototipo que van reemplazando a sus emulaciones. Así, la memoria RAM del sistema de desarrollo reemplaza al principio a la memoria RAM y ROM del prototipo, hasta que éste haya sido confeccionado y el programa haya sido depurado. A medida que va avanzando la implementación, las partes emuladas por el sistema de desarrollo van siendo ejecutadas en el prototipo.

En una última fase se completa el hardware del prototipo. El sistema de desarrollo emula entonces sólo el microprocesador a través del cable de unión, pudiendo grabarse la EPROM del prototipo con operaciones de E/S especializadas. Una vez que todo funcione perfectamente sólo resta sustituir el conector del emulador por el microprocesador del prototipo, y éste funcionará autónomamente.

En la figura de la página siguiente se ilustra un diagrama de los pasos comentados:



L O C A L I Z A C I O N D E A V E R I A S E N
S I S T E M A S C O N M I C R O P R O C E S A D O R

La filosofía de la reparación destinada a los sistemas basados en microprocesadores es similar a la de los diseños digitales estándar.

Una buena e importante ayuda para llevar a cabo las reparaciones es estar familiarizado con el sistema o circuito es cuestión. El conocimiento de la teoría de funcionamiento, el diagrama de bloques y el esquema eléctrico son la base imprescindible para comenzar a trabajar.

A continuación se comentan los problemas y las técnicas de reparación específicas de los sistemas con microprocesador.

PROBLEMAS EN LAS REPARACIONES DE SISTEMAS CON MICROPROCESADOR

Los problemas que surgen a la hora de las reparaciones de estos sistemas son muchos. Normalmente, gran parte del control reside en el software, por lo que es difícil seguir el flujo de la señal. Además, todo ocurre a una velocidad demasiado grande como para poder verlo en tiempo real. A diferencia que otros circuitos digitales o lógicos, un sistema microprocesador no puede ser parado mientras está en proceso para examinarlo. Las medidas han de tomarse mientras el microprocesador se encuentra en operación. Estas peculiaridades descartan como posibles instrumentos de reparación, o al menos reducen su uso, a la sonda lógica, el generador de impulsos; pero por otra parte elevan la utilidad del trazador de corriente, el osciloscopio, el analizador de firmas y el analizador lógico, puesto que estos instrumentos toman sus medidas con el circuito en actividad.

Otra característica del microprocesador es su estructura de buses, que implica más dificultades. Los datos e informaciones de las líneas de los buses suelen ser inestables o vacíos; la razón es que las salidas

pueden ser triestado, multiplexadas, conmutadas con transitorios,... En realidad, estas características no suponen un problema al sistema microprocesador en sí, pues éste es síncrono y sabe siempre cuándo las líneas del bus contienen señales estables. El analizador de firmas y el analizador lógico también lo sabe, ya que se les suministra la misma señal de reloj que está haciendo trabajar al sistema. Sin embargo, el osciloscopio no tiene esta capacidad. Proporciona poca información, pero es útil para ciertas cosas: actividad general, existencia de señales, niveles lógicos, diagramas de tiempo, conflicto de buses.

La estructura de buses permite conectar a un nudo muchos elementos. Si hay alguno defectuoso, encontrarlo resulta bastante complicado. El trazador de corriente es adecuado para este tipo de problema: el bus actúa como señal digital con realimentación y tiende a propagar los errores a través de circuitos en buen estado, para luego regresar a la fuente del fallo. El mejor tratamiento en este problema consiste en abrir el camino de realimentación cuando sea posible.

Con frecuencia se conectan a las líneas de los buses del microprocesador elementos muy complejos (LSI), difíciles de comprobar. El funcionamiento correcto de

tales elementos puede verificarse sustituyéndolos por otros chips en buen estado, o bien observando si la función que realizan para el sistema se lleva a cabo correctamente.

Los microprocesadores son máquinas secuenciales, esto es, el programa que en ellos corre depende de una larga secuencia de instrucciones y acontecimientos. Un solo bit incorrecto puede hacer fallar a todo el sistema y hacerlo completamente inútil. Por lo común, las causas de los errores en los bits suelen ser señales digitales de ruido o fallos en los bits de memoria; y son difíciles de detectar, pues cuando algo va mal, parece que todo el sistema funciona incorrectamente; encontrar la fuente del problema es todo un desafío.

Existen en el mercado docenas de microprocesadores diferentes, y miles de personas dedicadas al diseño de productos y técnicas de servicio y reparación de los mismos.

PROBLEMAS ESPECIFICOS DE LOS SISTEMAS CON MICROPROCESADOR

Señales de reloj.-

Una señal de reloj incorrecta puede dar origen a

fallos, aunque el sistema siga funcionando. Algunos microprocesadores son sensibles a la velocidad del reloj. Así, por ejemplo, una pequeña variación en su velocidad (demasiado deprisa) puede provocar el fallo total del sistema porque no se da tiempo a los distintos elementos para que realicen su trabajo, hagan sus operaciones, actualicen o estabilicen los datos después de una conmutación o transición.

Por otro lado, si el sistema funciona con demasiada lentitud, son las células de memoria de tipo dinámico las que pueden fallar. Estos problemas en la velocidad suelen aparecer cuando los circuitos de estabilización de frecuencia son del tipo R-C; siempre se debe recurrir, por tanto, a un cristal que nos dará un control más estable y exacto. No obstante, los cristales pueden fallar en su tercer armónico de oscilación, demasiado alto, originando una velocidad en la señal de reloj mucho más elevada que la prevista. Algunos microprocesadores necesitan incluso unas señales de reloj muy estrictas, multifase y sin solapamiento.

Las características de las señales de reloj de los microprocesadores se hallan en las hojas técnicas de datos de los componentes y pueden medirse o comprobar su buen funcionamiento usando osciloscopios y contadores de

frecuencia normales.

Señal de RESET en la conexión de la alimentación.-

La circuitería anexa al reseteado y conexión de la alimentación pueden también estropearse, y el sistema seguir funcionando. Un impulso de RESET inexistente, demasiado corto, o demasiado lento en transición, o con exceso de ruido, puede poner al sistema en marcha, estropeándolo.

También se pueden presentar problemas en la señal de RESET originados por las perturbaciones de la alimentación. Incluso cuando se emplean circuitos de Schmitt como entrada, los flancos lentos pueden ser causa de distorsión del tiempo de RESET de un elemento a otro. Esto hará que unos elementos se carguen antes que otros en la conexión de la alimentación, dando lugar a un comportamiento erróneo.

Una secuencia demasiado rápida en el circuito de alimentación ON-OFF-ON obligará a ponerse en marcha de nuevo a muchos sistemas; entonces se hace necesario incrementar el tiempo OFF para permitir el suministro de la alimentación y volver a poner en marcha los circuitos para, que se descarguen.

A pesar de estos posibles fallos del reset del sistema, éste puede seguir operando. Puede seguir corriendo el programa durante un corto periodo, y luego pararse o detenerse en un bucle poco importante del programa. El punto importante que se ha de tener en cuenta es que el sistema debe completar la secuencia de RESET en la conexión de la alimentación, para asegurar que han tenido lugar todas las operaciones de examen, control e inicialización de la puesta en marcha del sistema.

Los circuitos de RESET de la alimentación funcionan por lo general sólo cuando se conecta al sistema la alimentación por primera vez. En ese momento puede estudiarse el funcionamiento con osciloscopios con memoria o analizadores lógicos.

Interrupciones.-

Las líneas de interrupción con situaciones de ruido pueden hacer fallar al sistema. Un sistema puede seguir funcionando con una interrupción atascada, pero pasarla la totalidad o al menos parte del tiempo en atender a esa interrupción. Pueden además producir cambios o resultados aleatorios en el sistema, o bien hacer operaciones de entrada y/o salida en momentos inadecuados.

Cómo estudiar la actividad de una línea de interrupción? Fácilmente se puede hacer con una sonda lógica, osciloscopio o analizador lógico. Además, las interrupciones que sean asíncronas son controlables de forma manual, a veces, y podemos examinarlas realizando nosotros mismos su activación o desactivación.

Degradación de la señal.-

Los buses de datos, direcciones y control suelen ser largas líneas situadas todas juntas y en paralelo; esto las hace susceptibles de ciertas perturbaciones, originadas en líneas adyacentes de señal, u oscilaciones transitorias:

- Diafonía: se produce debido a un acoplamiento entre líneas.
- Problemas de transmisión: mala transmisión en líneas muy críticas, como las señales de reloj, o la habilitación de chips.

Cualquiera de estos motivos pueden ocasionar una falta de datos o señales de control, o bien una falsa toma de los mismos, situaciones muy difíciles de detectar. Las condiciones que pueden favorecer a la generación de este

tipo de problemas son diversas:

- Líneas de señales demasiado largas.
- Abuso en los márgenes de ruido.
- Abuso en los tiempos admisibles que indica el fabricante o diseñador.
- Adición al sistema de nuevas tarjetas para su ampliación.
- Condiciones ambientales de alta humedad.

Memorias.-

Una avería en memoria puede dar lugar a un fallo total del funcionamiento del sistema, o bien al almacenamiento de un dato erróneo. Tanto una situación como otra la puede generar un error en un solo bit. Estos fallos son detectables en el transcurso de los programas de autotest del propio sistema, que corren al conectar inicialmente la alimentación, a menos que dicho fallo impida correr a ese programa de autotest.

Si el sistema no dispone de este tipo de pruebas de verificación de RAM, y además no tiene conectado ningún procedimiento de comprobación de memoria, revisar su buen

funcionamiento es casi imposible. La única solución que queda entonces es sustituir el chip correspondiente, siempre que se sospeche que es la memoria la causa de la avería.

Un simple fallo en la zona de RAM destinada al stack hace que el sistema no funcione correctamente, aunque sólo exista un error pequeño. Imaginemos, por ejemplo, que el error está en un bit de un byte perteneciente a una dirección de retorno de un subprograma o rutina. Qué ocurriría cuando el microprocesador intentara retornar? Que saltaría a una dirección equivocada, siendo impredecible lo que ocurriría a continuación.

Sin situarnos en estos casos extremos, la avería en una RAM puede ser lo suficientemente leve como para que el programa siga corriendo con normalidad, a excepción de que se obtendrían resultados inciertos.

El fallo de circuitos de refresco de las RAM dinámicas es otra cuestión a tener en cuenta a la hora de diagnosticar averías en RAM.

También es posible la avería en una memoria ROM. Un solo bit malo estropearía todo el sistema, haciéndolo inútil: si, por ejemplo, se trata de un bit en una

instrucción, el código erróneo podría representar a otra instrucción, o bien a ninguna existente dentro del código del microprocesador. Tanto en un caso como en otro, los resultados pueden ser catastróficos.

La ROM puede comprobarse de forma conveniente y efectiva durante el programa de autotest en la conexión inicial de la alimentación, si el sistema se ha diseñado previendo estos programas de comprobación. Si no hay que usar otras técnicas; una de las más utilizadas consiste en hacer correr el programa del sistema, y con la ayuda de un analizador de firmas examinar los resultados o comparar las salidas de una ROM de prueba con las de otra que sepamos de antemano que funciona correctamente.

PROGRAMAS DE AUTOTEST

Gracias a la característica de programabilidad de los sistemas basados en microprocesador se pueden realizar programas de autotest para dichos sistemas. La misma ROM del sistema es capaz de contener un programa destinado a testearse a sí misma, a la RAM y al propio microprocesador. Comentemos esto con más detalle.

EXAMEN DEL MICROPROCESADOR.-

La forma más corriente de llevar a cabo un testeo del microprocesador es mediante la comprobación del buen funcionamiento de sus instrucciones, al menos de algunas de ellas.

Por ejemplo, para verificar la instrucción ADD se guardan en memoria ROM dos números, que actuarán de sumandos, y otro más que es la suma de ambos. Un programa hace que el microprocesador sume esos dos números y compare el resultado obtenido con el existente en la memoria. Si son iguales es señal de que el microprocesador ejecuta la instrucción ADD de forma deseable. Estrategias similares pueden emplearse para comprobar otras instrucciones, o el juego completo de ellas, con lo que tendríamos la certeza de trabajar con un microprocesador en buenas condiciones.

EXAMEN DE LA MEMORIA ROM.-

El procedimiento más común utilizado para llevar a cabo un examen de las memorias ROM consiste en una prueba de verificación por suma, llamada "checksum".

Cuando se programa la ROM se suman todas sus palabras, ignorando todos los arrastres que resulten. Esta suma se complementa y se almacena en la última palabra de la ROM, o en la primera; de esta forma, cuando todas las palabras se suman entre sí y se suman además al byte checksum, el resultado debe ser cero. Si no es cero al final de la suma total, quiere decir que algo falla en la ROM. Actualmente en la práctica se realiza un checksum para que el número resultante total sea un número cualquiera diferente de cero.

Pero el método de checksum no es infalible, o por lo menos no es del todo fiable. Detecta cualquier error de bit único y la mayor parte de los errores de bits múltiples, o sea, errores en varios bits. Sin embargo hay muchas combinaciones de dos o más errores que son posibles y que además se producen de tal manera que estos errores se anulan unos a otros, haciéndonos parecer que la suma total realizada es correcta. Así que cuando se emplee este método de detección de errores, si la ROM rebasa la prueba no quiere decir que haya una total seguridad en que dicha ROM esté en perfectas condiciones; sólo disponemos de cierto grado de certeza. En cambio, si no rebasa la prueba es signo de que algo anda mal con seguridad, aunque puede que no sea la ROM la que falle; podría fallar el microprocesador (instrucciones, registros internos,

señales del mismo), o cualquier otro elemento que de alguna manera deba ser utilizado durante la ejecución del programa de testeo.

EXAMEN DE LA MEMORIA RAM.-

La mejor forma de comprobar una memoria RAM consiste en escribir un dato en la memoria, leyéndolo de nuevo y verificando que lo leído es igual a lo escrito.

Existe un modelo muy común de dato de comprobación; se trata del llamado tablero de ajedrez (checkboard). En este modelo todos los bits escritos son alternativamente unos y ceros. Cuando se hayan comprobado todas las posiciones de memoria se repite la prueba pero al revés; donde antes se escribía un uno ahora se escribe un cero, y viceversa. Con esto se logra verificar si en cada célula de la RAM se puede escribir un uno y un cero. Existen otros patrones para la RAM, destinados a detectar ciertos fallos específicos en la misma.

No obstante, se ha de recalcar que este método no nos proporciona una seguridad del 100% sobre el buen estado de la RAM. Si, está claro que se ha comprobado que cada posición almacena correctamente la combinación 10101010, y la 01010101, pero nadie nos garantiza que suceda lo mismo

con la combinación 01111000. Incluso para verificar una pequeña RAM con una seguridad del 100% necesitaríamos mucho tiempo, pues tendríamos que comprobar en ella todas las combinaciones posibles en todas sus posiciones de memoria, y todas las secuencias posibles de estas combinaciones. Por esta razón, la garantía de las pruebas comentadas es inferior a las de una ROM.

Como conclusión, si una RAM supera el autotest sólo podemos decir que probablemente esté bien; si no lo supera sabremos que está mal.

EXAMEN DE PUERTOS.-

Dependiendo de qué haya conectado a un determinado puerto, y dependiendo de si éste es de entrada o salida, la verificación de un puerto puede realizarse de diversas formas. En general es posible hacerlo enviando un dato de salida a ese puerto y leerlo posteriormente; ambos datos, el escrito y el leído, deben ser idénticos.

Se trata de un procedimiento similar al comentado para las memorias RAM; por tanto, para tener cierto grado de certeza en el test habría que utilizar el método del tablero de ajedrez (escribiendo alternativamente ceros y unos), o bien realizar el examen con todas las

combinaciones posibles, tal como se indicó. Esta última opción, no obstante, resulta bastante lenta y no se aconseja su empleo.

EXAMEN DE OTROS CIRCUITOS.-

Para concluir con los procedimientos de testeo de elementos de un sistema basado en microprocesador, resulta interesante exponer una posible forma de verificar un convertidor analógico-digital, ya que este circuito forma parte del presente proyecto.

Podrían idearse varias maneras de llevar a cabo el testeo de un convertidor A/D. Se nos ocurre la siguiente: colocando unos conmutadores en las entradas analógicas sería posible desconectar las tensiones normales que han de ser convertidas durante la ejecución del programa; con la conmutación se conectaría a las entradas analógicas otros valores de tensión fijos, predeterminados por el diseñador y concebidos para el testeo del convertidor. Los valores binarios de dichas tensiones se encuentran guardados en memoria ROM.

Se realiza entonces una operación de conversión de estas entradas prefijadas y se compara el resultado

obtenido con el almacenado en memoria. Si son iguales es prueba de que el ADC ha funcionado correctamente.

Es conveniente permitir un pequeño margen de error entre el valor convertido y el valor grabado en la memoria ROM, para prevenir posibles variaciones en la tensión proporcionada por el circuito confeccionado para el testeo. Por ejemplo, el convertidor ADC0808, trabajando con una tensión de referencia entre 5V y masa, posee una resolución de 20mV; esto significa que una variación de 20mV en la tensión fija de testeo provocaría un resultado distinto al grabado en memoria.

Con este procedimiento tampoco tendríamos una certeza absoluta del buen estado del convertidor; para ello habría que realizar una conversión por cada valor posible de entrada y verificar que son correctos los resultados. Indudablemente se trata de un método difícil de llevar a cabo, por lo que es preferible hacer la comprobación de la forma simple indicada.

MULTIPLEXADO DE ENTRADA/SALIDA

Los teclados y displays frecuentemente están multiplexados, compartiendo los mismos circuitos de

exploración. Así es que una tecla trabada o defectuosa puede dar lugar a un fallo en el display; o bien de igual modo, una entrada incorrecta en el display origina un error en el teclado. Hay que tener muy en cuenta estas interacciones a la hora de diagnosticar una avería, pues podríamos pensar en un elemento en buen estado como causante del fallo.

INTERFACES

Es conveniente mencionar los problemas que pueden surgir en ciertas interfaces. Muchos sistemas basados en microprocesador suelen conectarse a otros a través de líneas de conexión externas. Estas conexiones pueden ser a corta distancia (locales) o a larga distancia, a través de la línea telefónica y por medio de modems. En este caso las líneas se encuentran expuestas a fuentes de interferencia eléctrica (relés, transformadores, motores, solenoides e incluso relámpagos).

Las interferencias electromagnéticas que emanan desde esas fuentes pueden ocasionar la transmisión de datos erróneos, sobrecargando los circuitos de interface, y especialmente en el caso de relámpagos o descargas eléctricas, produciendo grandes averías en los

componentes.

Normalmente, las líneas de salida de los circuitos conductores tienden a tener caldas de nivel más altas que lo usual, debido al potencial de las interferencias electromagnéticas y a las corrientes de alta transición que aparecen por la capacidad de los cables de interface.

DOCUMENTACION

A la hora de la reparación de una avería, los técnicos experimentados sólo precisan de la disposición de un diagrama de bloques correspondiente al circuito que van a examinar, y su esquema eléctrico. Sólo con esto son capaces de saber cuál es la función de cada elemento componente del circuito.

También les es útil disponer de un diagrama de flujo que explique de forma gráfica la teoría de funcionamiento del sistema. Para entender mejor en qué consisten estos diagramas podríamos compararlos con los flujogramas u organigramas utilizados en un diseño software: los organigramas muestran la filosofía del programa, y la secuencia de operaciones realizadas. Si construimos unos

diagramas similares para explicar la filosofía del hardware de un sistema, ayudaremos a clarificar las ideas sobre el mismo, y a dar una visión de conjunto; estos diagramas no están relacionados en forma directa con el hardware en sí, sino con su teoría de funcionamiento y su finalidad.

A continuación se expone una especie de guía a seguir para arreglar averías en los productos basados en microprocesadores. Se trata de algunas técnicas comunes.

EL RECONOCIMIENTO DEL PROBLEMA

Para empezar es necesario tener un conocimiento general del sistema o elemento estropeado, de forma que tengamos la seguridad de que está averiado. Se precisa saber qué es lo que hace y cómo lo hace, pudiendo existir distintos niveles de profundidad en su conocimiento.

La complejidad que el diseñador haya dado a su producto es muy diversa; un mismo equipo puede tener distintas formas de operar, unas más complicadas que otras, y la persona que va a reparar una avería en él ha de estar seguro de que el mal funcionamiento del mismo no se debe a un manejo o interpretación inadecuados. Otras veces se

intenta que el producto haga algo para lo que no se diseñó. Las limitaciones del diseño pueden encontrarse en el manual de funcionamiento y no constituyen una avería.

También podría ocurrir que se intentase hacer trabajar al sistema en situaciones no previstas por el fabricante; por ejemplo, el sistema posee un programa de ejecución que no cubre ciertas posibilidades. Esto sucede en productos recientemente fabricados, y tampoco constituyen una avería en sí. Cuando el usuario detecte una situación así deberá acudir al fabricante. Es posible también el caso de que realmente exista una avería, pero que no haya sido encontrada por uso insuficiente del equipo.

EL PANEL FRONTAL

Muchas veces un sistema microprocesador viene preparado con un panel frontal desde el cual podemos localizar averías, pues de él se puede obtener información para un diagnóstico.

En el panel puede haber interruptores, indicadores, entradas y salidas; es conveniente familiarizarse con todo esto. Así, por ejemplo, si los indicadores están apagados

cuando se conecta la alimentación, puede sospecharse de un interruptor defectuoso, o un fusible, o un cable, o cualquier conexión con la fuente de alimentación. Si un segmento de display está apagado, el problema podrá estar, probablemente, en el propio display, o en el circuito que lo gobierna.

EL MANUAL

Toda aquella información sobre las características del diseño, pruebas de examen y mensajes de diagnóstico que se puedan conseguir serán una ayuda para detectar averías. Normalmente esta información nos la proporciona el manual del sistema.

Hay que tratar de entender los circuitos y averiguar dónde está cada cosa. Se debe recurrir a la consulta de diagramas de bloques, esquemas eléctricos y a la teoría de funcionamiento que venga especificada en el manual. Tampoco es necesario un estudio exhaustivo y detallado de cada elemento, sólo lo suficiente para saber de qué va todo. Se ha de saber identificar el microprocesador, ROM, RAM, I/O, decodificador de direcciones, reloj, buses, control e interrupciones del sistema.

LAS AVERIAS DE CIRCUITOS INTEGRADOS

Un circuito integrado sigue durante su existencia una secuencia de pasos:

- El circuito integrado es fabricado en serie junto con otros circuitos.
- Después es vendido a un fabricante de sistemas o productos.
- El circuito integrado es insertado en una placa: entra a forma parte de un circuito en una tarjeta de circuito impreso, que junto con otras tarjetas configuran el sistema.
- A continuación entra en servicio para el resto de su vida.

No todos los circuitos integrados poseen una larga y buena vida. Los fabricantes de los productos estiman que un 2% de su producción es defectuosa, pero en la realidad este porcentaje es más alto. Una vez que el circuito integrado haya sido insertado en un equipo y éste se encuentre ya instalado y en manos del usuario, sustituirlo por otro resulta bastante caro. Es conveniente, por todo

esto, encontrar cuanto antes los circuitos integrados defectuosos en el ciclo de producción.

TIPOS DE AVERIAS

Podemos hacer una partición de las averías en dos tipos según el momento en que se producen:

- Averías en las líneas de producción.
- Averías de campo.

Llamamos averías en las líneas de producción a aquellas que aparecen en el momento de la fabricación. Se montan todos los elementos según el esquema y el equipo no funciona; esto puede deberse a la existencia de circuitos integrados defectuosos de fábrica, malas soldaduras, circuitos no cableados, ... En realidad, es probable que casi todo falle cuando se pone en marcha un producto en la fábrica.

Sin embargo, cuando un producto ha salido de fábrica en buenas condiciones y ha funcionado durante un tiempo en manos del usuario, una avería que sobrevenga después ya no puede deberse a las mismas razones que antes; se llaman averías de campo.

Instrumentos automáticos de comprobación.-

Existen en la fábricas equipos especializados de comprobación y búsqueda de averías, puesto que de otra forma llevar a cabo un examen de cada tarjeta del volumen total de la producción sería bastante lento. Utilizando probadores automáticos de tarjetas y pruebas fijas se consigue minimizar el tiempo de localización de fallos, de forma económica y fiable. No obstante, estas pruebas no tienen un 100% de fiabilidad. A veces tarjetas que han sido aprobadas por un tester o un programa de ordenador son defectuosas, debido a deficiencias en el propio programa de prueba. La credibilidad de estos aparatos aumenta con el desarrollo tecnológico.

COMPROBACIONES SENCILLAS

Hay tanta probabilidad de que un fallo se produzca en algo sencillo como en algo complicado de encontrar. Una de las zonas más propensas a fallar es la fuente de alimentación, así como una avería fácil de encontrar y solucionar. Una tensión inadecuada aplicada al circuito puede hacer que este funcione mal; por tanto, se recomienda comprobar el voltaje aplicado al sistema antes de seguir adelante.

Otra de las primeras cosas que se debe verificar es la parte mecánica. La instalación de tarjetas, sus cables de conexión, hilos rotos, componentes flojos, son aspectos a tener en cuenta.

AVERIAS COMUNES EN LAS LINEAS DE PRODUCCION

Hay muchas averías originadas en las líneas de producción que son fácilmente detectables por simple examen visual de los circuitos. Por ejemplo, es fácil detectar puentes o interruptores defectuosos, o bien soldaduras frías, componentes estropeados.

Los fallos más comunes en la producción son las soldaduras y los cortos entre pistas de las placas de circuitos impresos; éstos pueden arreglarse simplemente con un cuchillo afilado.

En la fase de producción de los circuitos, la inserción automática de circuitos integrados es capaz de originar la curvatura de sus pines, dando lugar a una conexión abierta entre el pin del circuito integrado y la placa de circuito impreso (o el zócalo, si el circuito integrado está montado sobre uno); al menos podría dar lugar a conexiones

intermitentes. Estas situaciones son a veces difíciles de detectar, puesto que los pines presentan la apariencia de estar perfectamente soldados o colocados. Asimismo es conveniente revisar los conectores de las tarjetas.

Para la confección de prototipos es idóneo el montaje con wire-wrap. Presenta muchas ventajas frente a las simples placas de prueba, pues disminuyen considerablemente el ruido que se puede generar y da un mayor grado de seguridad en cuanto a las conexiones. Sin embargo se ha de vigilar, por su parte inferior, que las patillas de los zócalos no están produciendo ningún corto, pues éstas son lo suficientemente largas como para originarlo si se les tuerce un poco.

Se ha de tener cuidado también con la inserción de circuitos integrados de catorce pines en zócalos erróneos de dieciséis, y con los posibles acoplamientos de señal debido a la proximidad de cables y pistas.

FALLOS MECANICOS

Estos fallos son del tipo de falsos contactos, cables desconectados o partidos, pistas rotas, paquetes de

resistencias y circuitos integrados cerámicos resquebrajados, terminales curvados de wire-wrap, conectores sucios, relés enclavados, circuitos integrados mal encajados en sus zócalos, o con pines partidos.

TECNICAS GENERALES DE REPARACION DE AVERIAS

Cambio de tarjetas.-

El intercambio de tarjetas estropeadas por otras en buen estado es muy usual en la reparación de averías, siempre que se tengan a mano sus duplicados. El riesgo de esta técnica de sustitución está en que se dañen las tarjetas buenas al quedar afectadas por sobrecargas eléctricas que estropearon las anteriores. De todas formas, una regla a cumplir siempre es desconectar la alimentación cuando se hace una instalación o reemplazo de tarjetas.

Si es posible adquirir un producto idéntico al que ha fallado, las comparaciones de uno con otro nos da mucha información útil en las situaciones en que no se sabe muy bien de qué problema se trata.

Si un componente con zócalo resulta sospechoso, se debe

golpear ligeramente o presionarlo, para ver si se trata de un mal contacto. Uno de los últimos elementos del que se debe sospechar es el propio microprocesador, pero normalmente se suele reemplazar el primero. Actualmente, el porcentaje de fallos del microprocesador es pequeño, mas son difíciles de verificar por su complejidad. Es por ello que se les suele sustituir en primer lugar, al igual que otros chips LSI que acompañan al microprocesador.

Pruebas de sobrecarga.-

Esta prueba es también llamada "test de stress", y es aplicada en casos de fallos intermitentes o marginales. Con ella se puede conseguir la eliminación de estos fallos, o su deterioro temporal.

Las tarjetas se sobrecargan térmicamente mediante calentamiento (secador, pistola de aire) o enfriándolas (aerosol refrigerante); y eléctricamente mediante aplicación de un voltaje distinto. Quizás la sobrecarga térmica sea más efectiva, pues el calentamiento o enfriamiento se le puede aplicar a un componente individual y detectar el fallo con más precisión.

Tocando los elementos de la tarjeta se puede localizar el que se ha calentado más que los demás. El calentamiento

de un elemento de la tarjeta por encima de otros iguales es sospechoso; incluso puede llegar a quemar el dedo. No obstante, hay elementos que se calientan considerablemente cuando están trabajando, y sea algo normal en ellos.

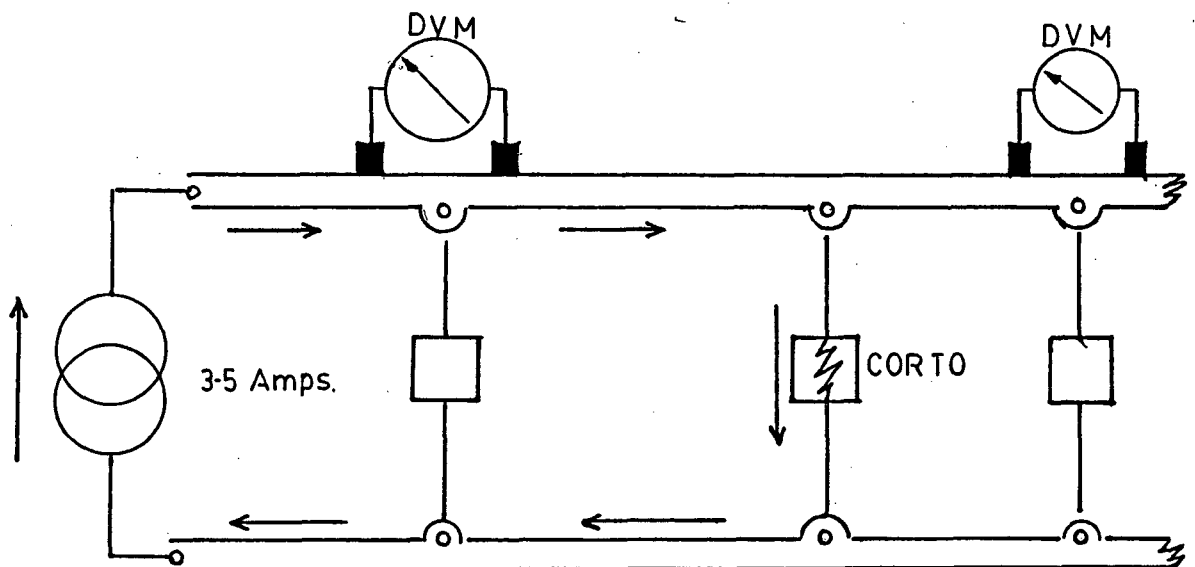
Cortos en la fuente de alimentación.-

Existen varios procedimientos para detectar cortos en la fuente de alimentación. En primer lugar se debe averiguar en qué tarjeta se produce el corto (si el sistema es multitarjeta), y se realiza sacando una tarjeta cada vez hasta encontrar la que evite el corto.

Veamos tres técnicas usadas para localizar un corto en una tarjeta:

- Aplicando un generador de impulsos que inyecte corriente a través de las dos líneas en corto. El trazador de corriente ayuda a seguir la corriente hasta el corto. También con el trazador se puede examinar la corriente a través de un condensador y compararla con la de otro condensador idéntico para ver si existen niveles idénticos. Así se pueden detectar condensadores de acoplamiento cerámicos cortocircuitados.

- La localización de cortos en los buses de alimentación se realiza suministrando una corriente alta, de unos tres o cinco amperios, al corto. Debemos asegurarnos de que no excede el voltaje de alimentación que exista normalmente, y que se mantiene la misma polaridad. Con un voltímetro de alta resolución (DVM de 0.01mV) podemos examinar el camino de la corriente hasta el corto, estudiando las caídas de voltaje durante el recorrido del bus de alimentación. Se apreciarán caídas de voltaje sólo en las partes del bus que forman el corto. La figura siguiente ilustra este método.



- Una técnica muy curiosa para localizar cortos en el bus de alimentación es la de congelar la tarjeta que tiene el corto, a una temperatura de unos -10°C . Después se le aplica una corriente de tres a cinco amperios, y según se caliente se irá descongelando el camino de la corriente, haciéndose visible este camino.

AISLAMIENTO DE LA AVERIA

Cuando habiendo utilizado las técnicas más sencillas de detección de averías no se ha conseguido localizarla, es momento de recurrir a otros métodos y hacer un examen más profundo. Aquí ya influye la destreza del reparador, su experiencia y el conocimiento que tenga del producto.

Se recomienda seguir el procedimiento de mitades, muy común en los algoritmos software de búsqueda de un elemento. En este caso el elemento a buscar es aquel que produce la avería, y empezaremos tomando un punto central de la tarjeta para examinar qué mitad de ella funciona correctamente y cuál no. Este método es bastante intuitivo, es decir, quizás todos lo hayamos utilizado alguna vez sin tener conciencia de ello.

Si es posible, el punto de partida para la búsqueda debe ser aquel de separación entre parte analógica y parte digital. Los circuitos analógicos tienen mayor porcentaje de fallos que los digitales, generalmente; las razones son su mayor velocidad, carga, sensibilidad, los reducidos márgenes de seguridad de sus componentes, necesidad de mayor exactitud y ajuste...

Una fuente de ruidos puede ser originada por la posible interacción eléctrica del reloj y las líneas del bus de alimentación TTL con los circuitos analógicos.

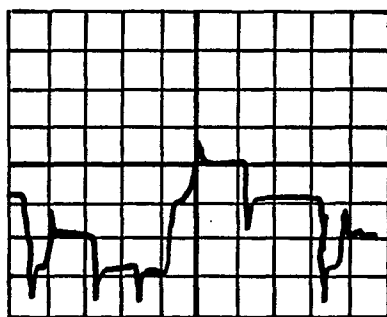
AVERIAS DIGITALES

Una avería en la parte digital ha de ser tratada, en primer lugar, comprobando la actividad de las distintas señales. Se examinan las señales de reloj, las líneas de los buses, los pines de habilitación y las líneas de control, y cualquier inactividad en ellas es un posible problema.

Los circuitos integrados pueden fallar muchas veces por la apertura de las conexiones de los cables dentro de la cápsula. Si una salida del cableado interno se abre, el

pin de salida quedará suelto, y probablemente la sonda lógica indicará un estado de alta impedancia.

Si una entrada queda abierta, una o más salidas del circuito integrado serán erróneas. Si cualquiera de estas líneas está conectada a un bus triestado, puede originar un conflicto de bus, esto es, en una misma línea desembocan varias salidas activas en el mismo momento. Un trazador de corriente puede ayudar a encontrar este fallo; o bien mediante un osciloscopio, donde se detectaría anomalías en los niveles lógicos de las líneas del bus. En la pantalla del osciloscopio podría aparecer una gráfica como la de la figura en caso de la existencia de un conflicto de bus.

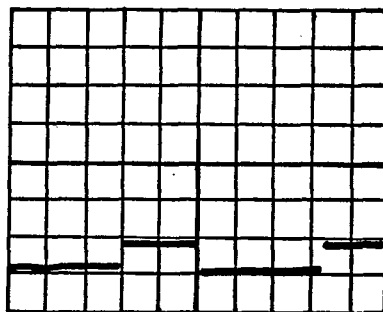


CONFLICTO
DE BUS

También podríamos llegar a pensar que existen errores en líneas buenas del bus, por presentar niveles anormales o inadecuados cuando se han desconectado todos los elementos del bus.

Es corriente el caso de que un pin de un circuito integrado se encuentre cortocircuitado a masa, debido, por ejemplo, a una mala entrada en el diodo de protección del chip. Normalmente aparece como un nivel bajo, que se puede detectar con una sonda lógica.

En un osciloscopio se lograría una gráfica como la de la figura próxima: siempre un nivel cero, excepto cuando se aplica un uno a la línea correspondiente, produciendo una señal de unos pocos milivoltios. Con el trazador de corriente es fácil localizar los pines de entrada cortocircuitados.



CORTO EN UNA PUERTA

TECNICAS DE AISLAMIENTO

Después de localizar un pin de entrada o salida sospechoso de ser la causa del problema se debe proceder a su aislamiento dentro del circuito. Esto se consigue extrayendo el estaño que une el pin al circuito impreso, con ayuda de un desoldador de vacío. A continuación se dobla el pin, de forma que no toque ningún otro sitio. Con un tester se puede comprobar que dicho pin está totalmente aislado.

La forma de aislar bloques digitales de sistemas basados en microprocesador depende del propio sistema, de su arquitectura eléctrica y mecánica. Cuando el núcleo principal del sistema pueda trabajar libremente en bucle abierto, se puede verificar el funcionamiento de esa parte principal y controlar la actividad de los buses.

Con una tarjeta de ampliación consistente en interruptores de líneas de señal y bus se pueden seleccionar señales entre la tarjeta y el resto del sistema, con lo que se logra apartar los circuitos de realimentación y los buses enclavados del sistema principal.

El examen de las líneas de un bus es posible realizarlo también de la siguiente forma: para saber si todas las líneas del bus están en buen estado se mide la resistencia a tierra de estas líneas, con el aparato desconectado. Todas las líneas deben tener un valor de resistencia aproximado, así que si se detecta que una de ellas ofrece una resistencia bastante diferente a las demás, es señal de que puede existir un problema. Si hay dos líneas que ofrecen la misma resistencia (baja), las dos han podido cortocircuitarse entre sí. De todas formas será conveniente revisar el esquema para ver si los componentes o circuitos conectados a estas líneas del bus son la causa de las diferencias halladas.

BUCLAS DE REALIMENTACION

Encontrar una avería en un bucle de realimentación digital es muy difícil, pues los errores se propagan en ellos mismos. Un bucle de realimentación con una señal de salida incorrecta vuelve a enviar esta señal a la entrada para producir más salidas falsas. Si se abren los caminos de realimentación evitamos que las señales erróneas de salida regresen a la entrada; hecho esto ya se podrán suministrar unas entradas controladas y examinar las salidas que se generan. Muchas veces es difícil producir

estas entradas y producir además la operación correcta en el circuito; en tal caso será mejor comparar el producto bajo examen con otro idéntico en buen estado y trabajando en las mismas condiciones.

Comprobar si es un circuito integrado el que está estropeado se puede hacer estudiando las señales generadas por el que se supone mal; quitarlo, colocar otro en buen estado, examinar si produce las mismas señales. Si no es así el circuito integrado estará defectuoso. A veces, la sustitución de un elemento por otro es la solución más cómoda, o la única.

P A R T E I I

E S T U D I O D E U N S I S T E M A D E
C O N T R O L D E E S T A B I L I D A D

D E S C R I P C I O N D E L S I S T E M A

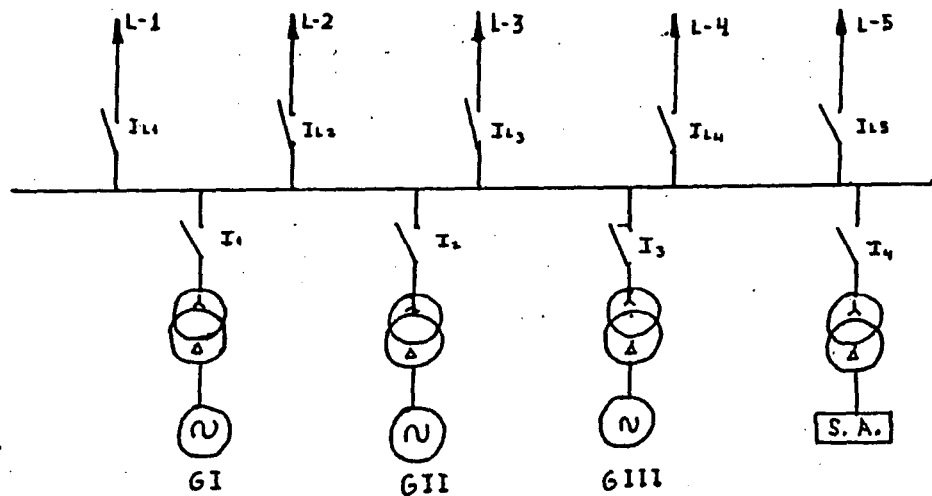
F I N A L I D A D

La existencia de fallos en el método actual empleado para la conservación de la estabilidad en las centrales eléctricas es la razón del nacimiento de este sistema de control.

El método utilizado, basado en relés de mínima frecuencia, resulta algo burdo por cuanto que no ofrece la actuación precisa ante casos de pérdida de la estabilidad. En principio, lo ideal para una central eléctrica es no caer nunca en un estado inestable; es lógico que éste constituya el objetivo fundamental de los mantenedores de la planta.

Partimos de un esquema concreto de central eléctrica. Nos referimos en todo momento a una planta formada por

tres grupos que trabajan en paralelo; las cargas acopladas a ellos son en total cinco.



ESQUEMA GENERAL: LINEAS Y GENERADORES

Las características del método basado en relés son las siguientes:

Se establecen varios niveles de actuación, fijados a distintas frecuencias. Cuando un generador se encuentra sobrecargado, esto es, se le exige más potencia de la que puede suministrar, comienza el fenómeno de pérdida de sincronismo. La frecuencia y la tensión normales en el grupo disminuyen, y la consecuencia final es el desacoplo

del mismo por actuación de su propia protección a base de relevadores.

Si un generador se desconecta de la red, inmediatamente la potencia que suministraba pasará a los grupos que quedan en servicio, sumándoseles a la potencia que ya ofrecían desde antes. Probablemente también esta potencia sobrepase sus límites; así, de forma análoga al primer generador, los restantes en servicio también se desacoplarán.

Antes de que esta circunstancia se dé, se supone que deben actuar los relés de mínima frecuencia.

Los relés, tal como se ha indicado, se accionan dependiendo de los valores de frecuencia recibidos. Si un generador comienza a perder el sincronismo se produce una disminución en su frecuencia.

Los relés tienen la característica de poder responder a un cierto valor de frecuencia, o umbral. Si la frecuencia del grupo rebasa un límite inferior impuesto (considerado como nivel crítico), el relé actuará desconectando de la red una o varias líneas asociadas a él.

Si además se coloca otro relé de mínima frecuencia en

el sistema, cuya frecuencia umbral esté por debajo del anterior, podrá ser accionado otro conjunto de líneas, en caso de que la frecuencia del grupo aún siga disminuyendo.

La descarga de líneas alivia a los generadores, pues se reduce la demanda y se permite el restablecimiento de los grupos.

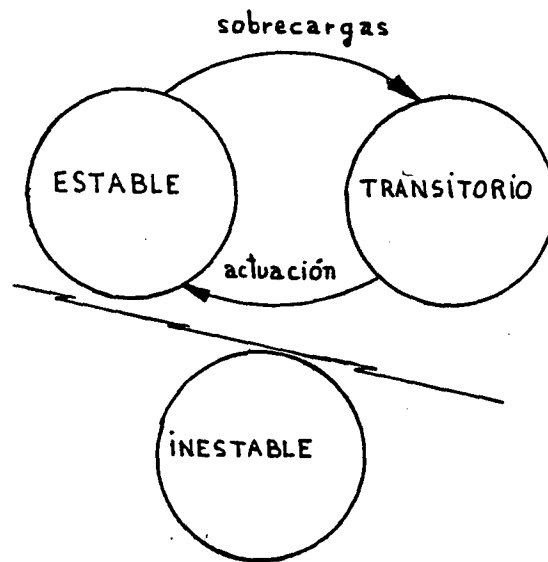
No obstante este método presenta graves inconvenientes, resultando poco efectivo en ciertas ocasiones. La lentitud de respuesta es uno de ellos. También son poco selectivos: en el momento de descargar líneas siempre se verán afectadas en primer lugar aquellas asociadas al relé con mayor umbral de frecuencia, pudiendo ser una descarga injusta o inapropiada.

Por otra parte, es necesario esperar a que los grupos pasen de su estado estable a otro de pérdida de sincronismo antes de que los relés actúen; no interesa, lógicamente, que los generadores alcancen la inestabilidad. Es más, no siempre se consigue recuperar el estado estable nuevamente, produciéndose la calda de la central.

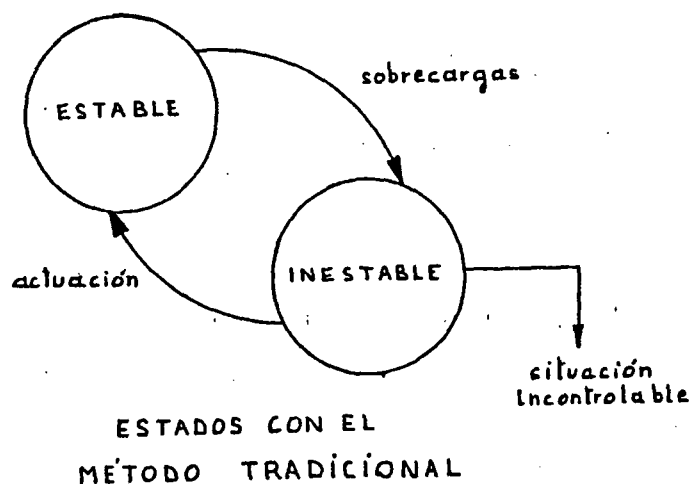
El objetivo de este sistema es precisamente evitar que se produzca tal estado. Lo que se intenta es controlar la

situación en cada momento, saber qué está pasando en cada elemento de la central durante cualquier instante, y actuar cuando se detecte un estado transitorio de estabilidad hacia inestabilidad. La actuación se realizará de tal forma que desde el estado transitorio el sistema vuelva al estable, sin llegar nunca al inestable.

Gráficamente lo anterior se expresa mediante los siguientes esquemas:



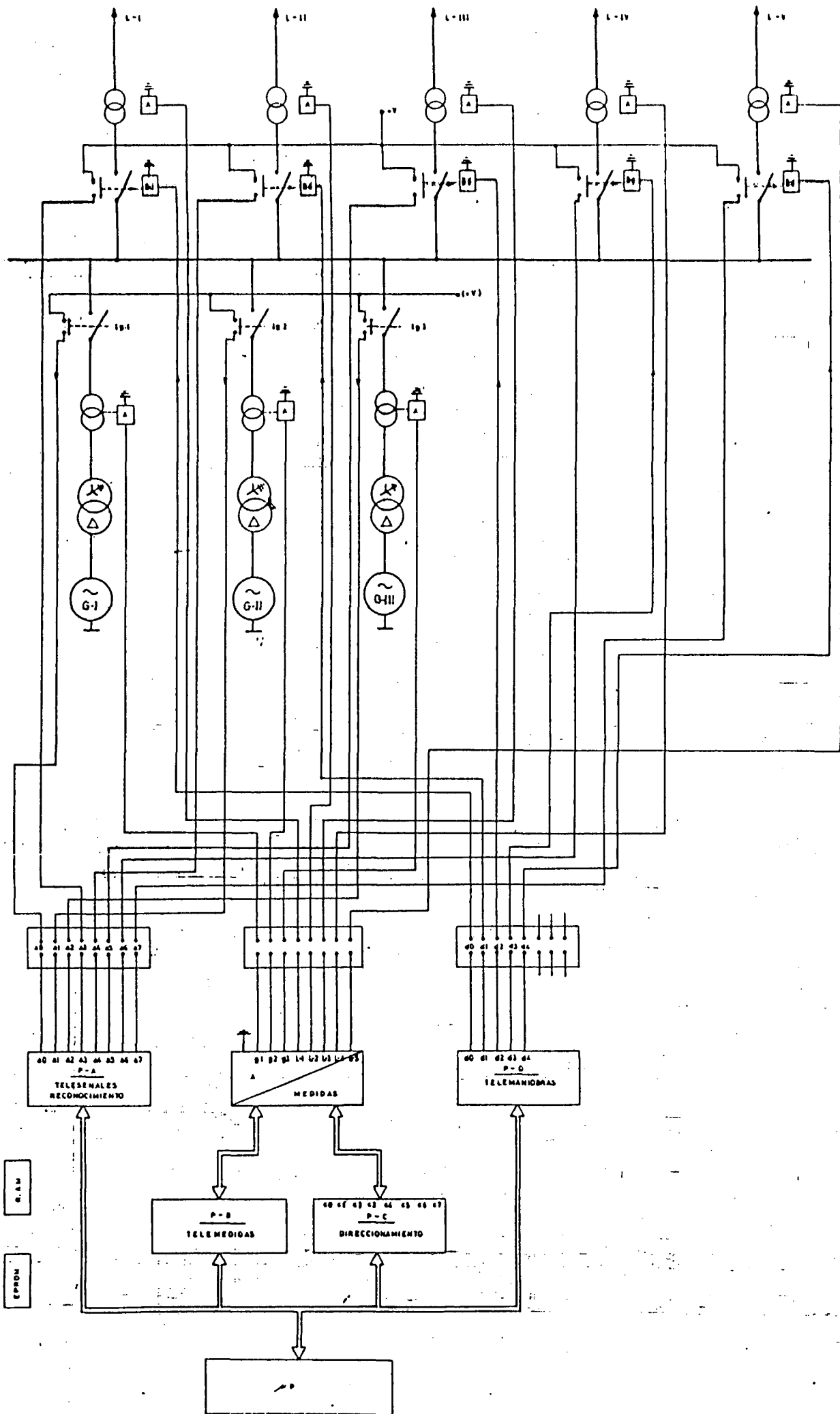
ESTADOS POSIBLES CON SISTEMA DE CONTROL



Con un sistema basado en microprocesador existe la posibilidad de tomar los datos necesarios sobre los consumos y generaciones, datos sobre el estado de apertura o cierre de los elementos, además de procesar dichas informaciones para, a partir de los resultados obtenidos, actuar de forma conveniente.

Una ventaja más sobre el método de los relés se consigue a través de la comunicación del sistema de control con un ordenador para el intercambio de información y la presentación en pantalla de los valores asociados a los elementos de la central en cada momento. El uso de microprocesador hace posible esta mejora.

CONTROL DE LINEAS Y GENERADORES



H A R D W A R E

El sistema de control diseñado está formado por tres módulos principales: CPU, toma de datos-envío de órdenes (E/S), y transmisión al ordenador. Cada una de ellas se detalla a continuación.

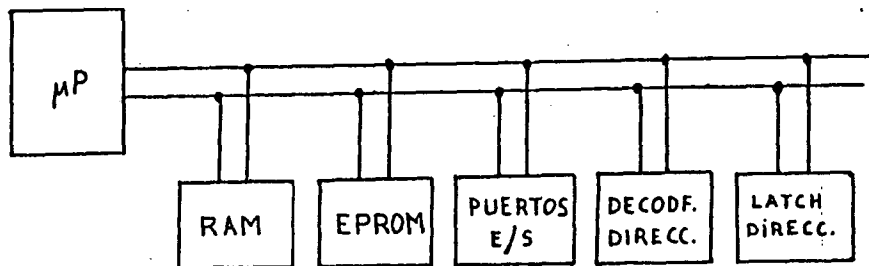
* El módulo de CPU lo constituyen los siguientes elementos: un microprocesador, una memoria EPROM de 4K con el programa de ejecución, dos integrados de memoria RAM de 256 bytes cada uno, varios puertos de E/S, dos temporizadores programables, a los que se suman circuitos auxiliares como un decodificador de direcciones y un latch de direcciones.

En este módulo se realizan procesamiento de señales, cálculos y toma de decisiones con la misión de controlar la central. Los integrados utilizados para ello son:

- el microprocesador 8085.
- el latch de direcciones 8212.
- el decodificador de direcciones 74138.

- la memoria EPROM de 4K 2732.

- dos integrados 8155 (RAM, puertos de E/S y timers).



ESQUEMA MÓDULO DE CPU

* El módulo toma de datos-envío de órdenes es el que tiene como función testear los valores de potencia de consumo y generación en cada momento, y pasarlas a la CPU para su procesamiento. Al mismo tiempo se envían las órdenes necesarias para la conservación de la estabilidad.

La toma de datos se divide en dos tipos:

- recepción de valores de potencias de los generadores y líneas.

-, recepción del estado de éstos en la red, o sea, el

estado OFF (desacoplamiento) u ON (acoplamiento) de cada elemento.

Los valores de potencia son suministrados por la central en forma de corriente. A través de unos convertidores analógico-analógico el valor de intensidad que circula por las líneas y generadores es traducido a una escala inferior; se trata de una muestra del valor real adaptada para su utilización en el sistema microprocesador.

Las corrientes que se nos proporciona varían dentro de los límites de 4 mA a 20 mA. Los 4 mA corresponden a una potencia real nula (si se recibe de una línea o generador este valor de corriente será prueba de que dicho elemento se encuentra desconectado de la red). El límite superior, 20 mA, será detectado cuando se alcancen los siguientes valores de potencia:

- un 110% de la potencia nominal en los generadores.
- un 120% de la potencia nominal en las líneas.

Tanto para unos como para las otras, la potencia nominal se considera de 9400 KVA, o su equivalente, 7500 Kw. Trabajar por encima de estos valores se considera situación de alarma. Cuando un grupo presenta una potencia

del 110% del nominal ya estamos ante un estado crítico, sobre el que se debe actuar.

El 110% de la potencia nominal es el límite superior admisible en un generador. Una sobrecarga mayor del 10% podría causar la inestabilidad, por lo que cuando tal magnitud sea alcanzada el sistema actuará desconectando líneas para aligerar carga.

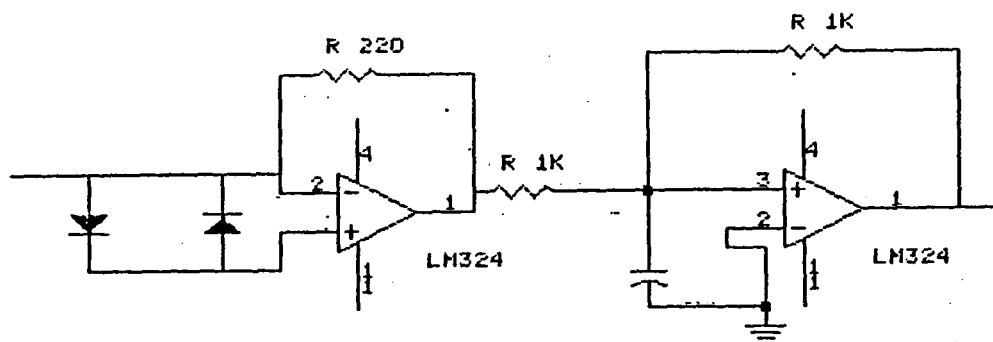
En definitiva, los operarios de la central suministran unas corrientes de 4-20 mA por cada generador y cada línea, corrientes que serán proporcionales a las potencias generadas y consumidas. Nuestro sistema las toma a través de las entradas analógicas de un convertidor analógico-digital.

El conversor utilizado es el ADC0808, que posee ocho entradas analógicas multiplexadas, y una salida digital de 8 bits. Direccionándolo de las ocho maneras posibles nos da los ocho datos digitales, que luego son introducidos en el módulo de CPU a través del bus de datos.

El convertidor sólo acepta tensiones como entrada, por lo que las intensidades deben ser pasadas a tensión para su aplicación al convertidor. La forma en que esto se lleva a cabo es mediante amplificadores operacionales y

algunos otros componentes anexos. La relación es tal que 20 mA de entrada al operacional se transforman en 4,4 V de salida, dirigidos hacia las entradas analógicas del convertidor.

El esquema eléctrico que consigue esta transformación se ilustra en la figura:



El amplificador operacional 1 actúa de convertidor corriente-tensión. La resistencia de 220 ohmios hace que una corriente de 20 mA a la entrada se considere a la salida como:

$$20 \text{ mA} \times 220 = 4,4 \text{ V}$$

La salida del operacional 1 es negativa y necesita un inversor en cascada para conseguir la tensión positiva necesaria. Los diodos y condensadores actúan como protección.

Los operacionales utilizados son los integrados en la pastilla LM324; cada pastilla posee cuatro circuitos operacionales; en total se emplean cuatro pastillas, proporcionando 16 amplificadores.

El convertidor ADC0808 está cableado en el prototipo de tal forma que la CPU accede a los resultados de las conversiones como si de ocho posiciones de memoria se tratara. Si se desea mayor información sobre este aspecto, se recomienda acudir al apartado "Estudio de algunos integrados utilizados" de este proyecto.

La tensión de referencia de conversión, de valor 5 V, ha de ser muy exacta e invariable. Para ese fin se utiliza un regulador de tensión, el LM723.

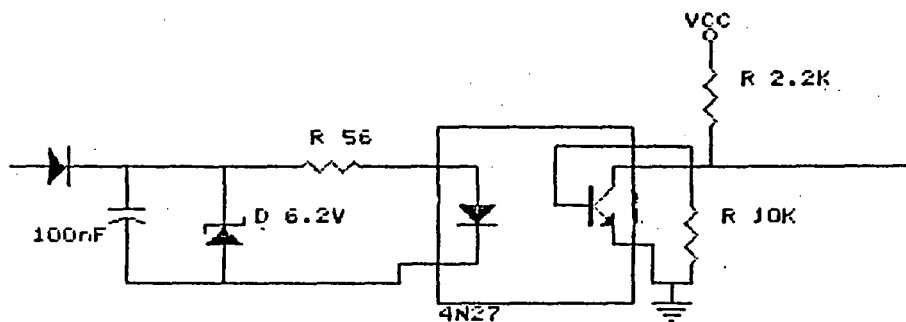
La señal de reloj de 51 KHz aplicada al convertidor para controlar la conversión se consigue por división de la señal de reloj del 8085; esta división se lleva a cabo mediante el timer incorporado en uno de los integrados 8155, concretamente el enumerado con 1 en los esquemas.

La señal de final de conversión está puenteada con la patilla RST 6.5 del microprocesador. De esta forma cada nueva conversión se realizará por la aparición de

interrupciones hardware.

En cuanto al estado ON-OFF de cada elemento, sólo se necesita un bit para su reconocimiento. Dichos estados son también proporcionados por los operarios de la central, y llegan a la CPU a través de un puerto de entrada de ocho bits (un bit por cada elemento).

Se ha realizado una separación de masas entre las señales llegadas desde la central (aportando la información ON-OFF) y las aplicadas al sistema de control en sí. Esta separación se lleva a cabo mediante los optoaisladores 4N27. Se emplea una circuitería anexa a los optoacopladores con el fin de generar la excitación precisa.



La aplicación a la entrada del circuito de una tensión de 5 V o de cero voltios origina respectivamente una

salida hacia el puerto de un nivel lógico 0 ó 1. Los cinco voltios de entrada corresponderán a un elemento desconectado; y viceversa, la aplicación de cero voltios equivale a un elemento conectado.

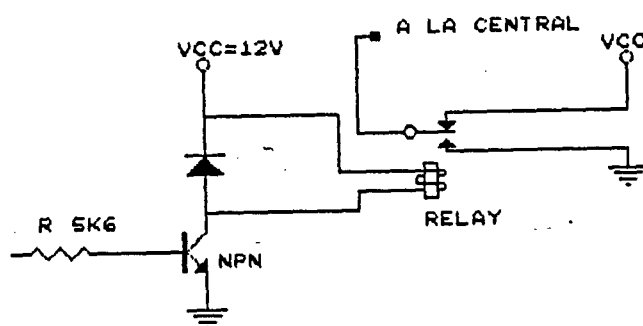
Uno de los circuitos integrados 8155 se utiliza como entrada de estos estados 0 y 1, concretamente el puerto A de la segunda 8155, que posee ocho bits. Son las entradas llamadas PA0-PA7.

Dentro del módulo de toma de datos-envío de órdenes queda por explicar en qué consiste el envío de órdenes y cómo se lleva a cabo.

Se llama envío de órdenes a los datos mandados desde el sistema a la central que tienen por objeto hacer conmutar las líneas acopladas a la red, lo que denominamos descarga de líneas.

Para llevar a efecto tal descarga se utiliza un puerto de salida de ocho bits, concretamente el puerto B de la segunda 8155, patillas PB0-PB7. De los ocho bits proporcionados por este puerto sólo se emplean las cinco primeras, PB0-PB4, ya que al sistema se le permite actuar sobre la desconexión de las cinco líneas, exclusivamente.

Los dispositivos de conmutación son los relés en miniatura de tres contactos (dos contactos fijos y un común móvil). El circuito de excitación de los relés se muestra en la figura:



La salida TTL del puerto B es aplicada a la base de un transistor, a través de una resistencia de polarización. El transistor es un 107B, y constituye la etapa de potencia, gracias a la cual es posible la activación del relé a partir de la señal TTL procedente del puerto.

Un nivel lógico 1 a la salida del puerto B pone en conducción al transistor, y el relé conmuta: significa que la línea está conectada. Un cero lógico, por el contrario, deja al transistor en corto, quedando el relé en reposo: significa línea desconectada.

* Por último pasemos a comentar el módulo de comunicación entre el sistema y el ordenador.

La comunicación se realiza a través de la USART 8251, cuya descripción se encuentra en el apartado "Estudio de algunos integrados utilizados" de este proyecto.

La forma de comunicación establecida es asíncrona, sin bit de paridad, con dos bits de stop, a una velocidad de 1200 baudios. Esta velocidad no conviene aumentarse por razones de fiabilidad de las transmisiones entre el sistema y el ordenador; además, los 1200 baudios son suficientes para las necesidades del sistema, no siendo precisa una mayor rapidez. La longitud de carácter es de siete bits; los datos son transmitidos en código ASCII.

A través del bus de datos del sistema se toman los datos en formato paralelo desde la CPU y se transmiten en formato serie por la patilla TxD de la 8251. De forma análoga, los datos recibidos en forma serie desde el ordenador son introducidos por la patilla RxD y pasados a formato paralelo antes de volcarse en el bus de datos.

La interface de la RS232-C con la USART se lleva a cabo por medio de los integrados 1488 y 1489. El 1488 transforma las señales TTL (entre +5 voltios y masa) en

señales comprendidas entre +12 y -12 voltios, preparándolas para la transmisión. El proceso inverso (paso de -12/+12 voltios a TTL) es realizado por el 1489, con las señales de recepción.

La 8251 emplea para su funcionamiento una señal de reloj igual a la del microprocesador por un lado (entrada CLK), y una señal de reloj para sus entradas TxC y RxC por otro. Esta última se obtiene por división doble de la primera. La señal de reloj del microprocesador se aplica al temporizador programable de la segunda 8155, y ahí es dividida por un factor 40. Esta señal es introducida por las patillas de reloj de transmisión y recepción de la USART, y a partir de una instrucción software de modo se vuelve a dividir entre 64. El resultado es una señal de 1200 Hz.

La comunicación es a tres hilos, también llamada de puente de lujo. Consiste en unir, a través de conectores canon de 25 pines, la señal de transmisión del sistema de control con la de recepción del ordenador, la señal de recepción del sistema con la de transmisión del ordenador, y por último un tercer hilo une las masas de ambos.

Las opciones de funcionamiento del sistema de control son dos:

- sistema de control autónomo, trabajando por sí solo, tomando datos, procesándolos y decidiendo órdenes a partir de sus conclusiones.

- sistema de control con el ordenador conectado (PC/XT), realizando las mismas tareas anteriores, pero además transmitiendo al ordenador los valores de corriente por las líneas y generadores, el estado de los mismos (ON-OFF), y aceptando en caso de que sea preciso un cambio de parámetros. Los nuevos parámetros de referencia se los proporciona el ordenador.

Para distinguir entre un tipo de funcionamiento u otro se utiliza un bit de uno de los puertos, concretamente el bit de orden cero del puerto A perteneciente a la primera 8155, patilla PA0. Si se detecta un uno en este bit del puerto de entrada, el sistema trabajará en modo autónomo; si se detecta un cero lo hará en modo de comunicación con el PC.

S O F T W A R E

El software del sistema se compone de dos partes:

- el software del sistema de control.
- el software del ordenador.

El primero, el software del sistema de control, es el programa destinado a correr en la EPROM de 4K del prototipo, y cuya misión es llevar la acción de control en sí.

El software del ordenador es el programa que debe correr en éste a fin de poder establecer comunicación con el sistema, recoger datos de él (estado de líneas y generadores, así como sus potencias); además se cuenta con la posibilidad de mandar desde el ordenador al sistema unos nuevos parámetros, refiriéndonos en concreto a las potencias nominales de los elementos considerados. Esto es, si en algún momento interesa cambiar las potencias nominales estándar grabados en EPROM, y hacer que el control se realice con otras potencias distintas, el ordenador puede enviar esos nuevos valores al sistema, y

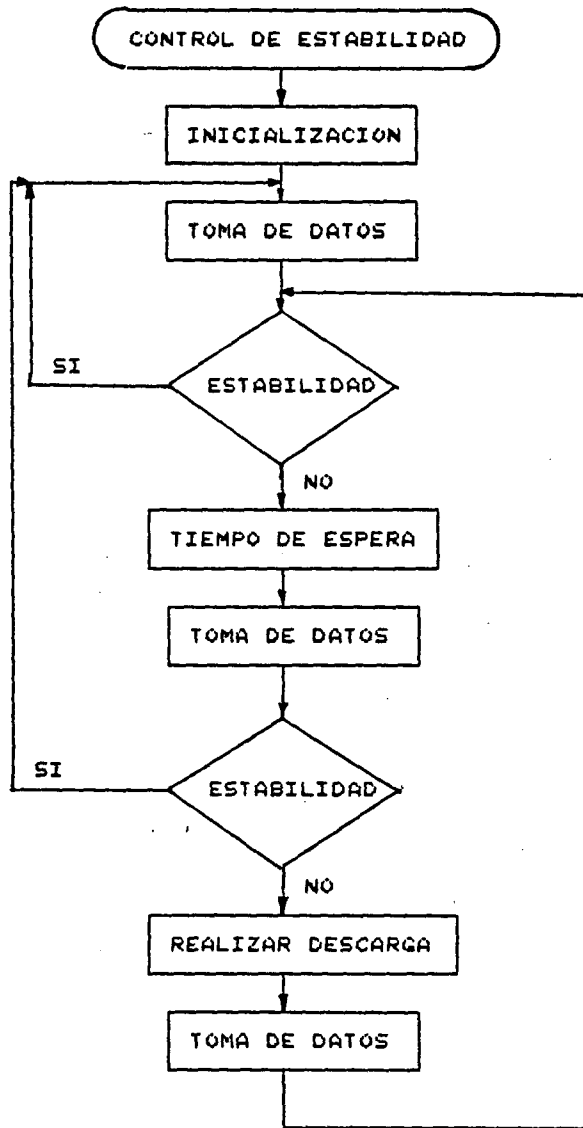
éste los aceptará y trabajará con ellos.

Veamos en qué consiste cada programa. No se realizará un estudio demasiado profundo, pues no es lo que se pretende aquí. Simplemente es necesario tomar una idea de qué hacen los programas, y la forma en que se lleva a cabo el control. Estos conocimientos pueden resultar imprescindibles a la hora de realizar cualquier mantenimiento o reparación del sistema de forma rápida y efectiva.

SOFTWARE DEL SISTEMA DE CONTROL

Como se ha indicado, se trata del programa del sistema microprocesador. La confección del mismo se ha realizado a través del sistema de desarrollo MDS-220 de Intel, concebido para el desarrollo de los microprocesadores 8080 y 8085. Los lenguajes de programación empleados son el PLM-85 y el ensamblador del 8085.

Se ha realizado en varios módulos que finalmente se han linkado. Un organigrama bastante general es el siguiente:



En primer lugar se realiza la inicialización, programando puertos, USART, dando valores a ciertas variables, y llevando a cabo cualquier otra acción necesaria.

Constantemente el programa testea los valores de potencia de los generadores para detectar si alguno está consumiendo más de su potencia nominal. Si este valor alcanza el 110% se considera una posible pérdida de estabilidad y se pasa a ejecutar un tiempo de espera.

La finalidad del tiempo de espera, llamado también tiempo de testeo, consiste en comprobar si el valor crítico detectado es un ruido o falsa alarma, o por el contrario se trata de una situación realmente preocupante.

Durante el tiempo de espera, los generadores restantes so vigilados para comprobar si continúan trabajando dentro de sus límites normales. Si alguno de ellos también está suministrando más del 110% de su capacidad normal, también se le asignará un tiempo de espera.

Tras finalizado este tiempo, el grupo que se encontraba en estado crítico vuelve a examinarse. Si la potencia de generación se restablece es indicador de que todo vuelve a la normalidad. Si se detecta que aún sigue suministrando

más del 110% habrá que actuar realizando una descarga de líneas. Después se vuelve a revisar cómo se encuentra el generador que estaba dando problemas; si su potencia ha disminuido por debajo del 110% el programa regresa al bucle normal de trabajo; si la potencia continúa siendo mayor o igual que el 110%, se provocará otra descarga.

Consideremos ahora unos organigramas más detallados, distinguiendo un programa principal y varias subrutinas.

El programa principal se denomina ESTABILIDAD, y a partir de él se deriva una serie de subrutinas.

* ESTABILIDAD:

Es el programa principal, encargado de la inicialización del sistema en cuanto a variables, programación de puertos, USART, timers, máscara de interrupciones, y cualquier otro aspecto necesario para establecer las condiciones iniciales.

Tras esto examina los valores de potencia de los generadores y actúa en consecuencia. Si las potencias sobrepasan el 110% del valor nominal, o llegan a él, se efectuará una llamada a la rutina de asignación de tiempo

de espera, RUTTESTEO. Cuando los valores son normales, el programa permanece en un bucle, y queda en él mientras no se sobrepase el 110% del valor nominal en ningún grupo. Las acciones que se mantienen dentro del bucle son:

- Toma de datos periódica sobre el estado ON-OFF de líneas y grupos, así como sus valores de corriente (proporcionales a sus potencias).
- Testeo de las potencias (o corrientes) recibidas correspondientes a los grupos, a fin de estudiar si estas corrientes aumentan, y si lo hacen de forma gradual o brusca, con el objeto de asignar los tiempos de espera adecuados, antes de optar por la descarga, en el caso de que se produzca un exceso de corriente mayor o igual al 10% de la nominal.
- Se realizan transmisiones periódicas al ordenador PC/XT para comunicar los datos recogidos sobre la central, en caso de estar trabajando en modo PC.
- Si se recibe la señal indicadora de cambio de parámetros procedente del ordenador, el sistema se prepara para recibir los nuevos valores (en caso de funcionamiento en modo PC).

Al comienzo del bucle del programa principal también se testea la forma de funcionamiento del sistema: autónoma o con conexión al PC.

* PCALFABETA:

Subrutina de comunicación con el ordenador PC/XT. Con ella se permite al ordenador enviar nuevos parámetros, mientras el sistema espera para recibirlos. Una vez metidos en memoria RAM, el sistema trabajará con ellos, en lugar de con los que posee grabados en EPROM.

Los parámetros a los que nos referimos son las potencias o corrientes nominales de los generadores. Así, si IN1, IN2 e IN3 son las corrientes nominales de los grupos I, II y III, respectivamente, se llama ALFA y BETA a los cocientes:

$$\text{ALFA} = (\text{IN2}/\text{IN1}) * 10;$$

$$\text{BETA} = (\text{IN3}/\text{IN1}) * 10;$$

Estos factores, junto con IN1 son enviados por el ordenador PC/XT al sistema de control. Los valores existentes en memoria EPROM y utilizados por defecto son:

ALFA = 10;

BETA = 10;

IN1 = 9400 KVA = 7500 Kw = 212;

El valor 212 corresponde a la traducción de 9400 KVA en el lenguaje del microprocesador. Es decir, 9400 KVA equivale a unos 18 mA en la señal suministrada por la central (mediante convertidores analógico-analógicos). Esto pasa por nuestro convertidor corriente-tensión y obtenemos unos 4,05 V. El convertidor analógico-digital da a partir de esta cifra un byte igual a 212 decimal.

* TRANSMISION AL PC:

Es la rutina de comunicación con el PC/XT encargada de transmitir desde el sistema los valores de corriente en los generadores y líneas, además de sus estados ON-OFF.

* RUTTESTEO:

Es la rutina a la que se accede cuando al menos un generador sobrepasa en un 10% su potencia nominal. En ella se asignan unos tiempos de espera dependientes de la forma en que el generador ha alcanzado tal sobrecarga.

Las formas pueden ser dos: incremental o brusca. Nos referimos a incremental cuando se produce de forma progresiva, gradual. Sin embargo, si se trata de un ascenso repentino decimos que se ha producido de forma brusca.

Tan pronto como algún generador sobrepase el 0,9 de su potencia nominal comienza a ser vigilado; se examina si se produce algún incremento y de qué tipo de incremento se trata.

Si el aumento de potencia ha sido brusco es probable que se trate de un ruido; entonces asignamos un tiempo que puede variar hasta 2,5 segundos, y testeamos tras él la potencia del generador. Si realmente era un ruido dicha potencia se habrá restablecido; pero si no lo era habrá que realizar una descarga inmediata.

Cuando el aumento de potencia sea gradual, lo más lógico es que el generador se haya ido sobrecargando poco a poco y ahora su potencia sea excesiva; no se tratará de un ruido, sino de un hecho preocupante. Cuanto más lento sea el incremento, menor será el tiempo de espera asignado, con un mínimo de algún milisegundo. Se provoca entonces una descarga y tras ella el sistema debe

recuperarse.

* RUTINA DE DESCARGA:

Se accede a ella cuando dentro de la rutina de testeo, RUTTESTEO, se haya tomado la decisión de provocar la desconexión de alguna línea. Su misión antes de la desconexión es seleccionar la línea según el siguiente criterio: se accionará aquella cuya potencia de consumo sea un 10% de la suma de potencias nominales de los grupos, o la inmediatamente superior a este valor.

Después de realizada la descarga se comprueba el valor de potencia del grupo que la motivó; si se ha normalizado el programa retornará al principal. Si no lo ha hecho, se descargará una línea más, hasta que la central se restablezca.

* TOMA DE DATOS:

A diferencia de las anteriores, a esta subrutina se accede por ocurrencia de una interrupción hardware. La toma de datos de potencia realizada mediante el convertidor A/D0808 se produce cada vez que la señal de

final de conversión (EOC) toma el valor lógico 1, indicando que la última conversión iniciada ha concluido y puede ser leído el valor correspondiente. En la subrutina de toma de datos se lleva a cabo esta lectura; además se inicia la próxima conversión correspondiente al valor de tensión aplicado en la entrada analógica consecutiva.

Cada vez que se produce la interrupción RST 6.5 (la patilla EOC está conectada a ella) se lee sólo una conversión y se dan las órdenes para el inicio de la siguiente, antes de retornar al punto del programa donde se dio la interrupción. Después de realizar la conversión de la entrada número ocho se lee el puerto de entrada indicador de los estados ON-OFF.

Con el reloj del convertidor analógico-digital a 51 KHz se consigue acceder a esta subrutina cada 1,64 ms. Esta constancia permite utilizar tal subrutina para una nueva tarea, la de servir de decrementador de los tiempos de testeo que se asignan en cualquier momento.

* RUTIN.ASM:

Para salvar los registros antes de la interrupción y recuperarlos después de ella se ha implementado una rutina

en ensamblador. Al producirse la interrupción el contador de programa salta a ella, guarda registros y llama a la toma de datos. Cuando ésta finaliza retorna al programa en ensamblador, se recuperan los registros, se habilitan nuevamente las interrupciones y se retorna al punto del programa en el cual se produjo la interrupción.

Los organigramas correspondientes y algunas notas aclaratorias sobre ellos se presenta a continuación.

Notas aclaratorias.-

- Estabilidad.-

Se llama $P(i)$ a un vector de tres elementos, $P(1)$, $P(2)$ y $P(3)$, que corresponden a unos valores relativos al tipo de incremento de potencia que ha tenido lugar en los grupos I, II y III, respectivamente.

$P(i)$ vale cero para aquel generador que no sobrepase el 90% de su corriente nominal. Cuando la corriente del generador exceda dicho valor se le asignará al $P(i)$ correspondiente un valor de

$$P(i) = 128 + P(i)$$

donde $P(i)$ es proporcional al exceso de la corriente $I_g(i)$ del generador por encima del 90% de su valor nominal. El valor máximo de $P(i)$ puede ser 138.

Para valores comprendidos entre 128 y 138, cuanto más cercano esté $P(i)$ al primer valor el incremento de potencia habrá sido más gradual; entonces se asignará un tiempo menor, en caso de que la potencia alcance el valor de 1,1 de su $I_{nominal}$. Cuanto más se acerque $P(i)$ a 138 el incremento de potencia habrá sido más brusco; en caso de que la potencia exceda el 1,1 de su nominal, el tiempo de testeo asignado será mayor.

- Rutina de testeo (Ruttesteo):

El vector $FLG(i)$ posee tres elementos, y son flags indicadores de si algún generador está mal. $FLG(i)$ a uno indica que el correspondiente generador ha sobrepasado el 1,1 de su potencia nominal.

- PCALFABETA:

El símbolo "R" se emplea para reconocer que se precisa un cambio de parámetros. Si el PC está conectado y se recibe este símbolo habrá que realizar la aceptación de los nuevos parámetros.

El símbolo "%" se utiliza para enviarlo al PC como

indicador de que se está preparado para aceptar los nuevos parámetros.

El símbolo "&" se usa como protocolo y confirmación de datos recibidos.

- Transmisión al PC:

El símbolo "\$" es para protocolo de comunicación. Reconocimiento de que el ordenador está preparado para recibir.

- Toma de datos:

La variable CANALDAC permite direccionar en cada momento la siguiente dirección de las ocho del convertidor analógico-digital.

S O F T W A R E D E L P C

Este es el programa que corre en el ordenador y se encarga de varias tareas. Se tienen las siguientes:

- Variar parámetros nominales del sistema.
- Transmitir estos nuevos parámetros y hacer que el sistema trabaje con ellos.
- Recibir desde el sistema los valores de corriente de líneas y generadores, así como sus estados ON-OFF.
- Visualizar en pantalla dichos valores, indicando además si alguno de ellos es no deseado (excesiva potencia). Esto se consigue haciendo flashear el valor correspondiente, o haciendo sonar una alarma.

Para la comunicación se utiliza el puerto número uno y se configura la transmisión a una velocidad de 1200 baudios, sin paridad, dos bits de stop, ocho bits por carácter.

Los parámetros pueden ser introducidos por teclado tomando como magnitud las KVA. El programa se encarga de transformar estos valores en otros adaptados para trabajar

el sistema y transmitirlos. Los parámetros variables desde el ordenador son las potencias nominales de los tres generadores, transmitidos como ALFA, BETA e Inominal.

La visualización en pantalla se efectúa mediante caracteres gráficos. La pantalla se dibuja en tres columnas: la primera contiene los valores actualizados de la potencia de los generadores y líneas. La segunda presenta los valores nominales de las potencias de los generadores. La tercera, el estado ON-OFF de cada elemento. Estos datos se van recibiendo uno a uno en forma serie por el canal de comunicación desde el sistema. Se van introduciendo en un vector $A(i)$ y posteriormente son presentados en pantalla.

Después de cada recepción de un paquete de valores la pantalla es refrescada y actualizada. Un paquete de valores de recepción consiste en un conjunto de nueve datos: tres generadores, cinco líneas, y un byte más con la información sobre el estado de los mismos.

Las transmisiones utilizan una serie de símbolos de protocolo. Así, antes de transmitir los valores de ALFA, BETA e Inominal se debe indicar al sistema microprocesador que tal hecho va a ocurrir, con la transmisión del símbolo "R".,

Una vez que el sistema está preparado para aceptarlos envía un símbolo "%" al ordenador. Este contesta con otro "%" seguido de los valores de ALFA, BETA e Inominal. Una vez recibidos, el sistema envía un símbolo "&" al PC como confirmación de buena recepción. En caso de que el PC no detecte este símbolo tras la transmisión se visualizará un mensaje de error de transmisión.

Existen dos posibilidades de acción sobre el programa del ordenador, con la introducción de dos controles: son los controles U y E.

- CONTROL-U: permite la congelación de la pantalla. Es decir, mientras el programa corre de forma normal los datos recibidos por el ordenador sobre corrientes y estados son visualizados en pantalla. Pulsando CTRL-U se logra que aunque se siga recibiendo nuevos paquetes de datos la pantalla no se actualice. Esto puede servir al usuario para el estudio de los valores en algún momento concreto.

Una vez pulsada la barra espaciadora, la pantalla volverá a su modo normal de refresco constante.

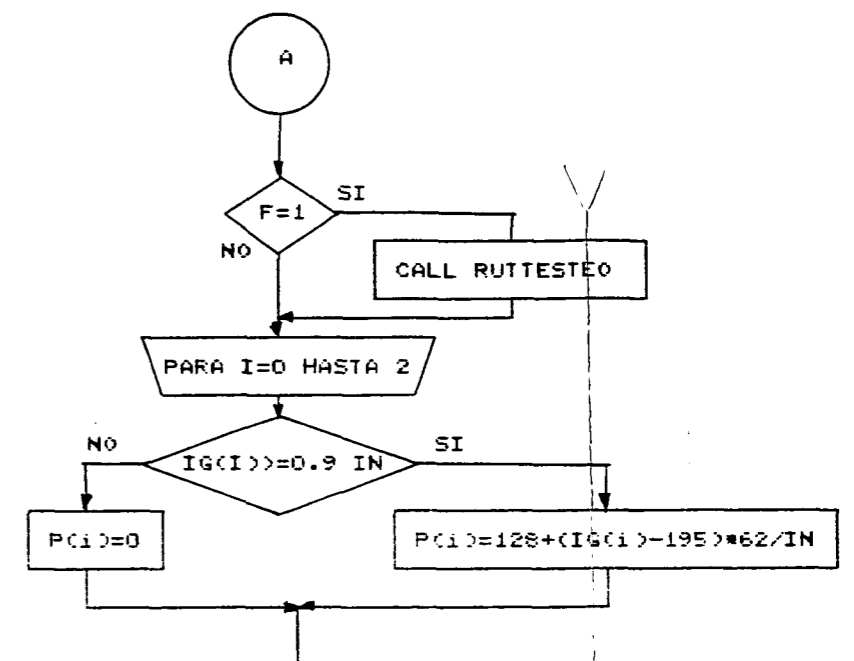
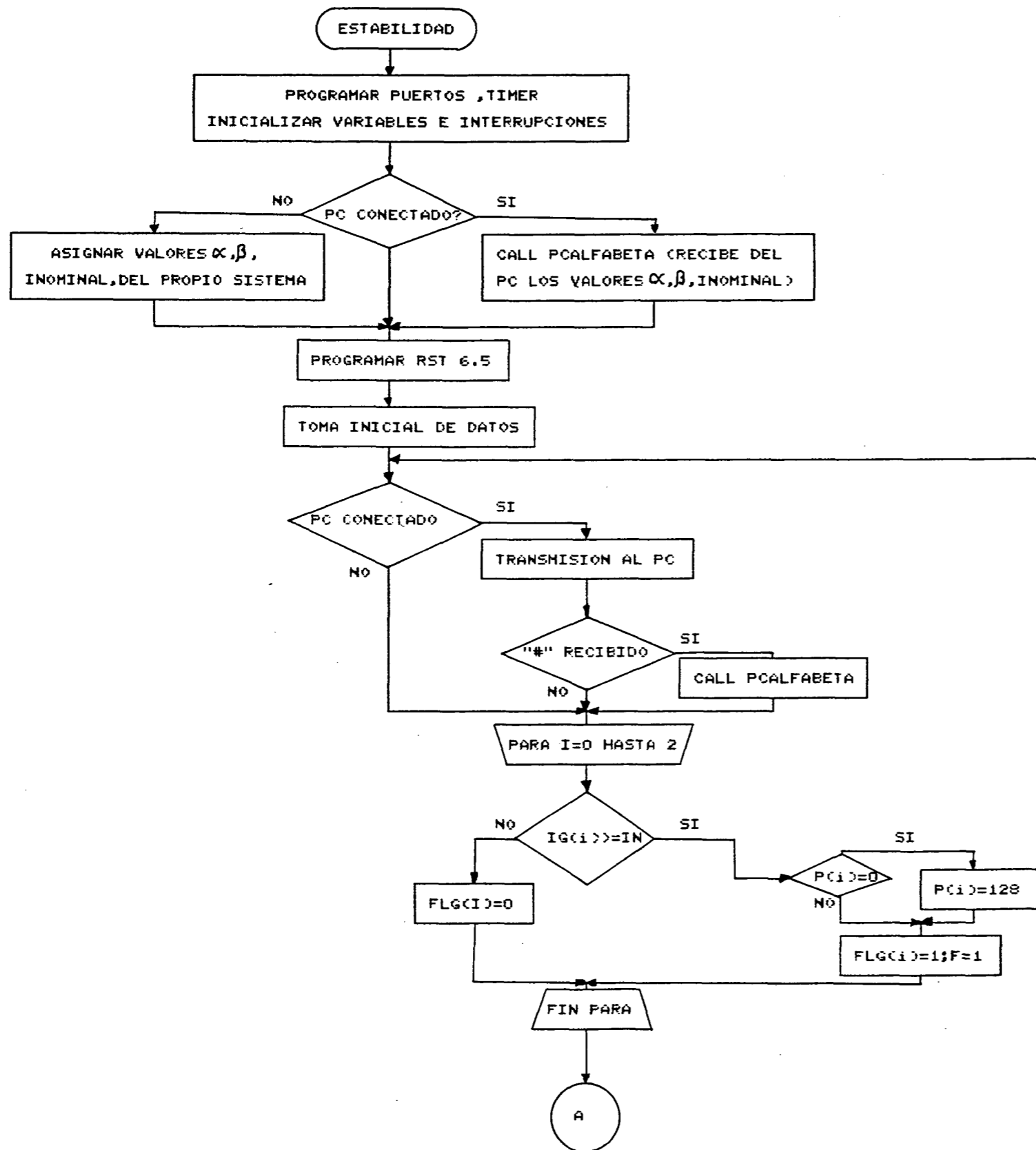
- CONTROL-E: posibilita el cambio de parámetros. Si en

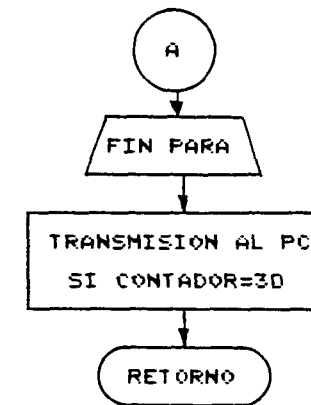
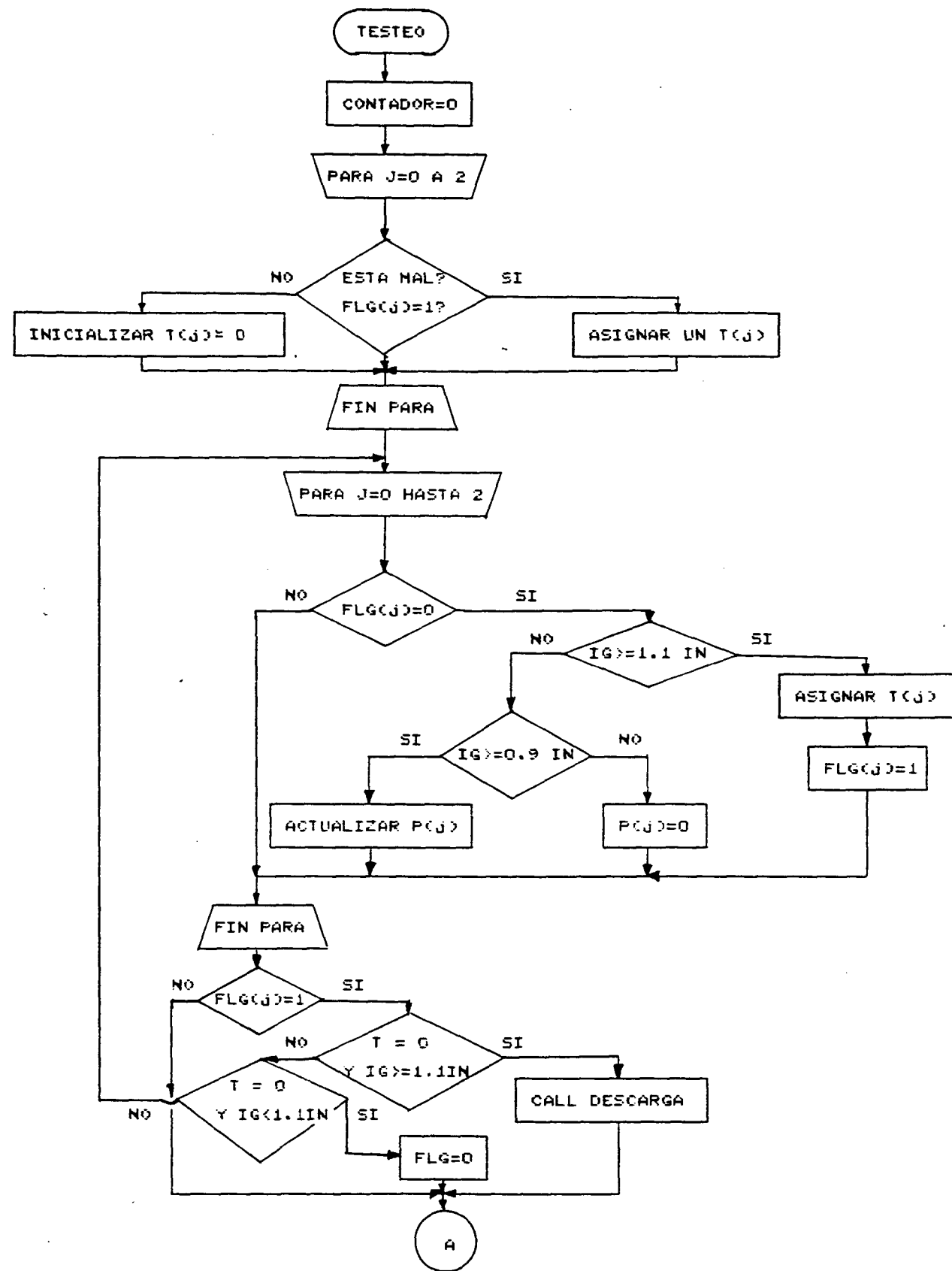
algún momento de la ejecución del programa el usuario desea cambiar los valores nominales basta pulsar CTRL-E y el ordenador interrumpirá la presentación de valores en pantalla y pedirá que sean introducidos los nuevos parámetros a transmitir. Tras la operación de comunicación al sistema el programa continuará con la recepción de datos y la presentación en pantalla de los mismos.

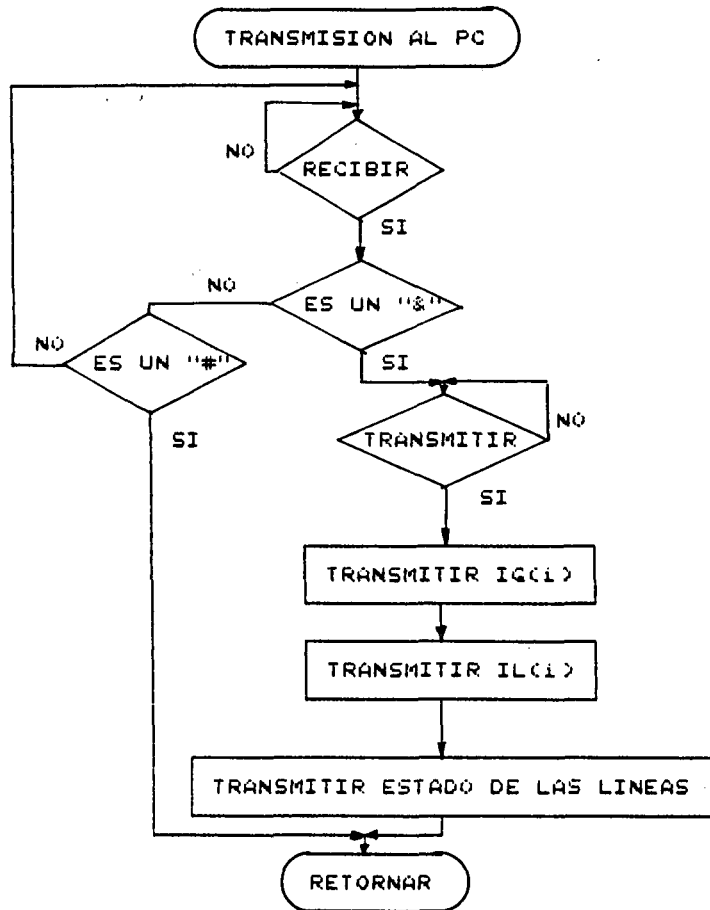
En la esquina superior derecha de la pantalla aparece siempre la sumatoria de las potencias en líneas y la de los generadores. Siempre tendrá que ser la sumatoria en los generadores igual a la sumatoria en las líneas, o al menos se permite un 2% de diferencia, pues hay que contar con los consumos de los servicios auxiliares. Si la diferencia es mayor que el 2% entonces se advertirá haciendo flashear estas sumatorias.

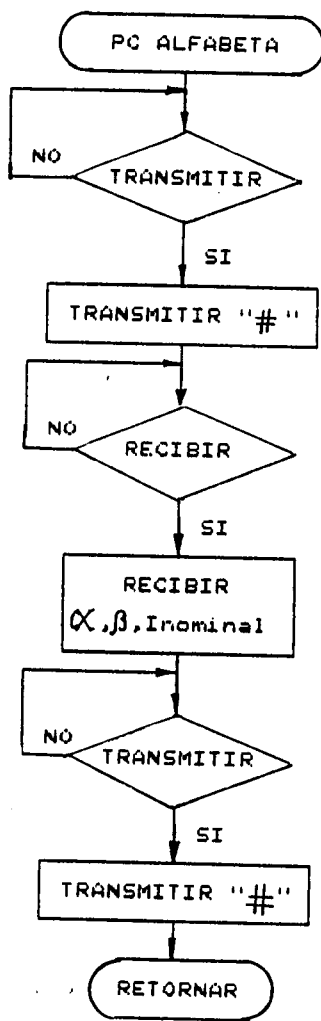
También se producirá el flasheo de aquellos generadores que superen la potencia nominal de suministro, así como sonará un pitido constante de advertencia, y continuará así hasta que el valor de potencia disminuya por debajo de la nominal. Otra circunstancia que producirá el flasheo de datos es el estado OFF de alguna línea o generador.

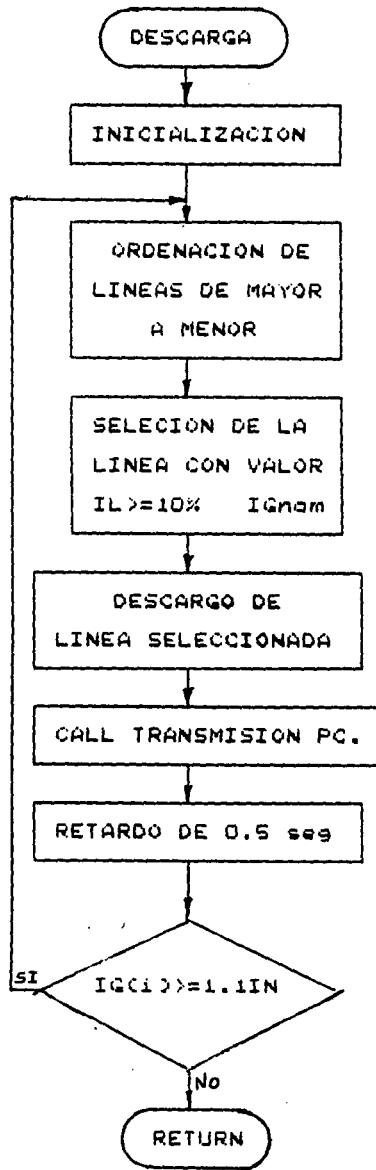
ORGANIGRAMAS

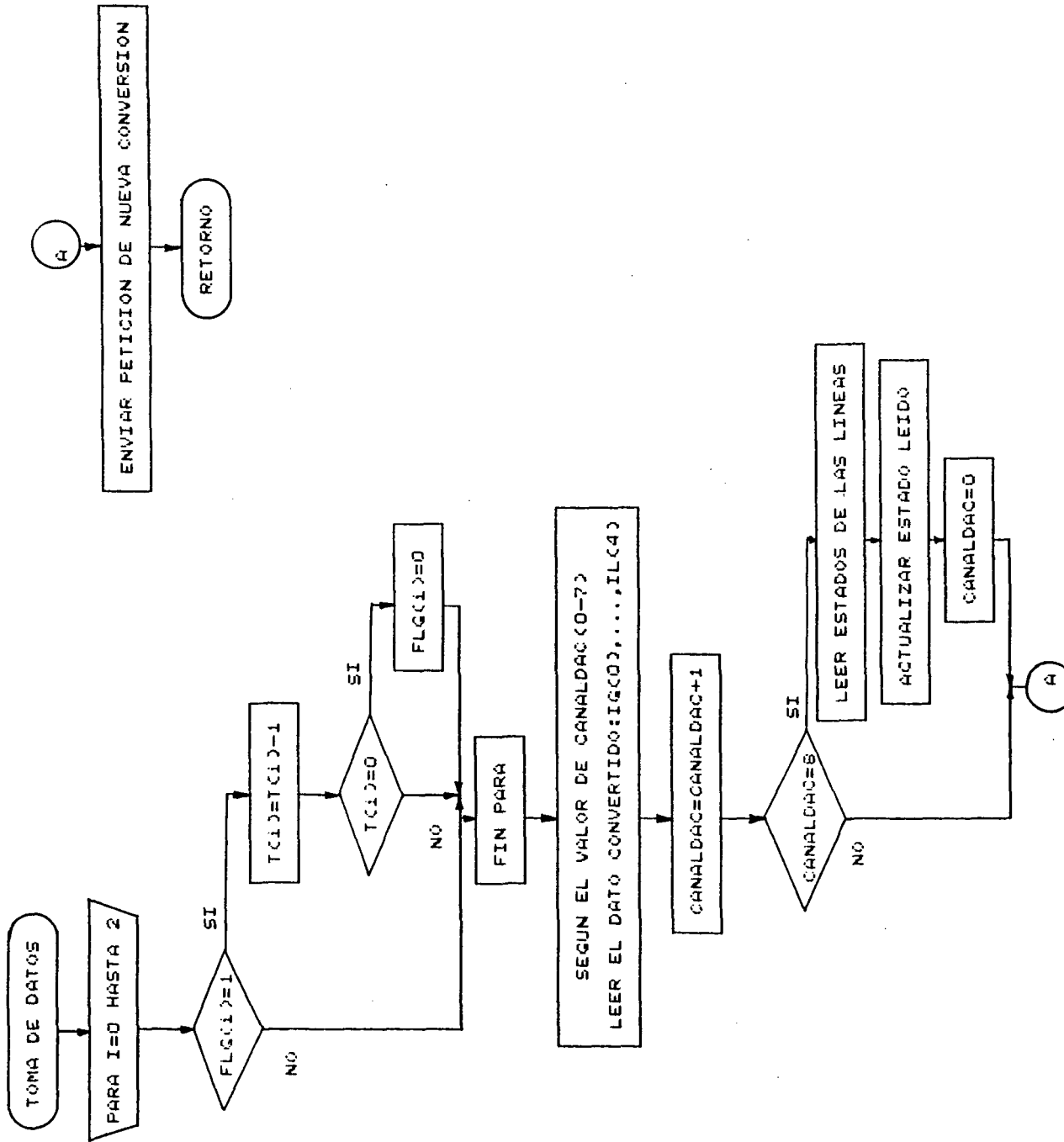


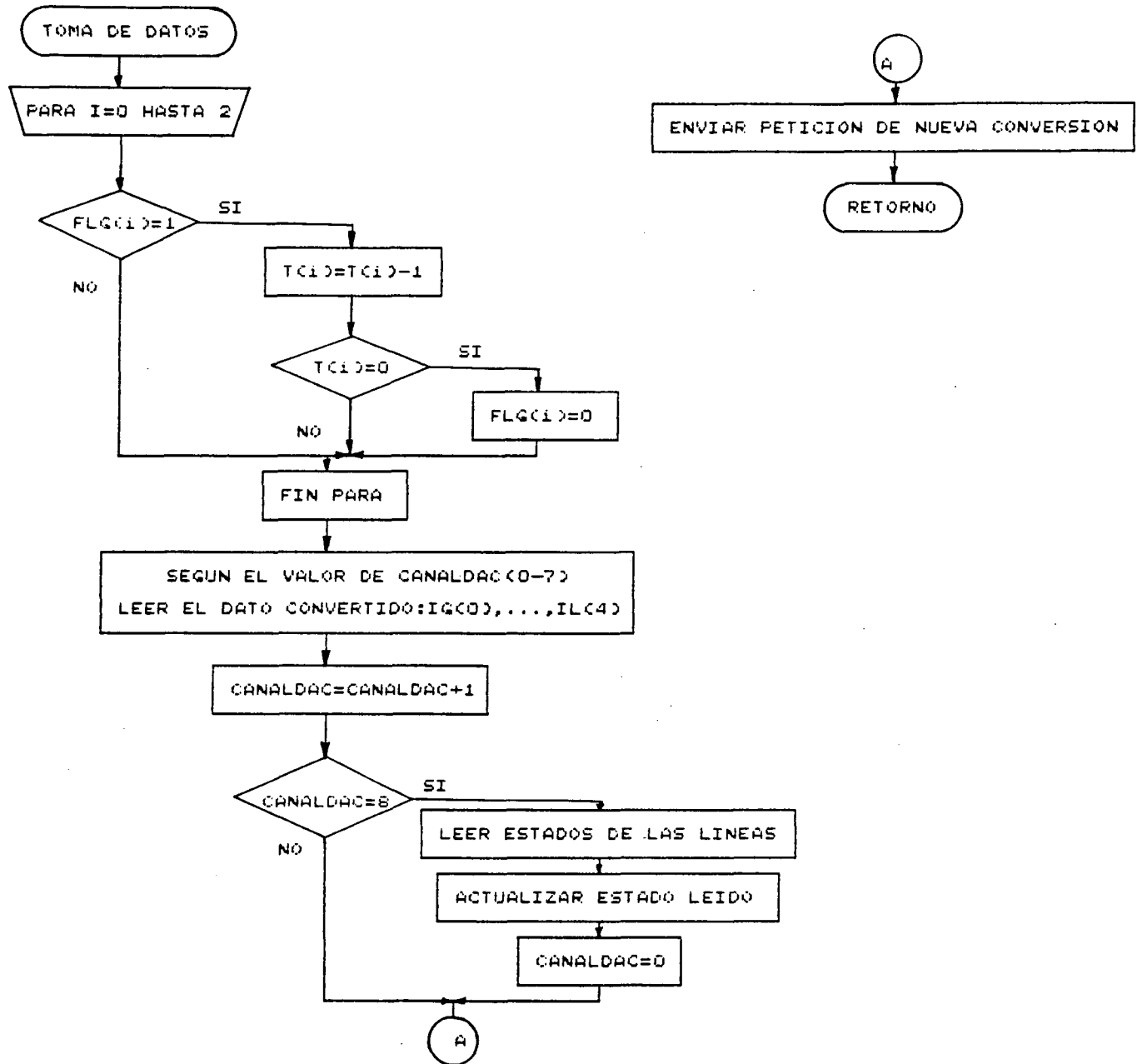


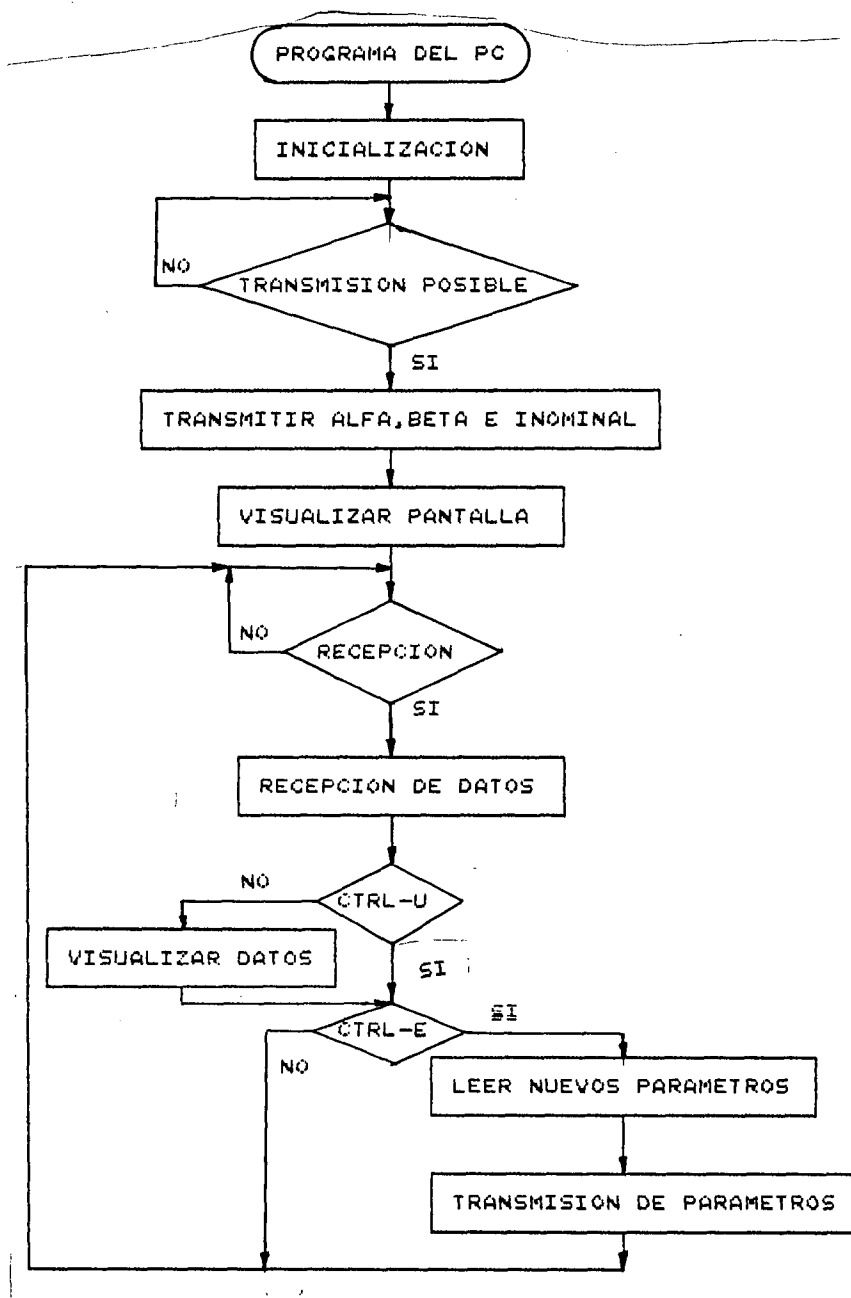












ORGANIGRAMA DEL PROGRAMA DEL PC

P R O G R A M A S

V E R S I O N M O D I F 3 P A R A
S I S T E M A M I C R O P R O C E S A D O R

P R O G R A M A S D E L S I S T E M A

M I C R O P R O C E S A D O R

ISIS-II PL/M-80 V3.1 COMPILATION OF MODULE ESTABILIDAD
 OBJECT MODULE PLACED IN MODIF3.OBJ
 COMPILER INVOKED BY: PLMB0 MODIF3.PLM WORKFILES(:F0:,:F0:)
 -DEBUG PAGELength(54) PAGEWIDTH(60)

```

/* Programa principal de control de es
-  tabilidad */
1      ESTABILIDAD:DO;

2      1      DECLARE IG(3) BYTE PUBLIC;
3      1      DECLARE IL(5) BYTE PUBLIC;
4      1      DECLARE FLG(3) BYTE PUBLIC;
5      1      DECLARE (LINDESCONECTADAS,CANALDAC,INIC,CO
-      R) BYTE PUBLIC;
6      1      DECLARE (ALFA,BETA,INOMINAL) BYTE PUBLIC;
7      1      DECLARE COMIEN LABEL PUBLIC;

8      1      TRANSMISION$AL$PC:PROCEDURE PUBLIC;
9      2      DECLARE USARDATOS LITERALLY '10H';
10     2      DECLARE USARCONTROL LITERALLY '90H';
11     2      DECLARE PORTB$RAM1 LITERALLY '42H';
12     2      DO;
13     3      LAZO4: B=INPUT(USARCONTROL) AND 02
-      H;
14     3      IF B<>02H THEN GOTO LAZO4;
16     3      B=INPUT(USARDATOS);
17     3      IF B='#' THEN RETURN;
19     3      IF B<>'#' THEN GOTO LAZO4;
21     3      LAZO6: B=INPUT(USARCONTROL) AN
-      D 01H;
22     3      IF B=0 THEN GOTO LAZO6;
24     3      OUTPUT(USARDATOS)='#';

25     3      DO I=0 TO 2;
26     4      LAZO1:B=INPUT(USARCONTROL) AND
-      01H;
27     4      IF B=0 THEN GOTO LAZO1;
29     4      OUTPUT(USARDATOS)=IG(I);
30     4      END;
31     3      DO I=0 TO 4;
32     4      LAZO2:B=INPUT(USARCONTROL) AND
-      01H;
33     4      IF B=0 THEN GOTO LAZO2;
35     4      OUTPUT(USARDATOS)=IL(I);
36     4      END;
37     3      LAZO3:B=INPUT(USARCONTROL) AND
-      01H;
38     3      IF B=0 THEN GOTO LAZO3;
40     3      OUTPUT(USARDATOS)=LINDESCO

```

```

-   NECTADAS;
41   3   TR10:           RETURN;
42   3           END;
43   2   END TRANSMISION$AL$PC;

44   1   RUTTESTEO:PROCEDURE (DIRPI) EXTERNAL;
45   2       DECLARE DIRPI ADDRESS;
46   2   END RUTTESTEO;

47   1   S$MASK: PROCEDURE (MASK) EXTERNAL;
48   2       DECLARE MASK BYTE;
49   2   END S$MASK;

50   1   DECLARE CONTROL$RAM1 LITERALLY '40H';
51   1   DECLARE CONTROL LITERALLY '90H';
52   1   DECLARE USARDATOS LITERALLY '10H';
53   1   DECLARE CONTROL$RAM2 LITERALLY '30H';
54   1   DECLARE PORTA$RAM2 LITERALLY '31H';
55   1   DECLARE PORTB$RAM2 LITERALLY '32H';
56   1   DECLARE PORTA$RAM1 LITERALLY '41H';
57   1   DECLARE PORTB$RAM1 LITERALLY '42H';
58   1   DECLARE TIMERL$RAM2 LITERALLY '34H';
59   1   DECLARE TIMERH$RAM2 LITERALLY '35H';
60   1   DECLARE TIMERL$RAM1 LITERALLY '44H';
61   1   DECLARE TIMERH$RAM1 LITERALLY '45H';
62   1   DECLARE IN LITERALLY '212';
63   1   DECLARE P(3) BYTE;
64   1   DECLARE (W,B,I,ED) BYTE;
65   1   DECLARE ADC(8) BYTE AT (2000H);

66   1   TOMA$INICIAL: PROCEDURE;
67   2       DO;
68   3           INIC=0FFH;
69   3           CANALDAC=00;
70   3           ADC(CANALDAC)=00;
71   3           ENABLE;
72   3       CUCO: IF INIC=0 THEN DO;
74   4           LINDESCONECTADAS=1FH;
75   4           OUTPUT(PORTB$RAM2)=LINDESCONEC

-   TADAS;
76   4           RETURN;
77   4       END;
78   3       TI10: GOTO CUCO;
79   3       END;
80   2   END TOMA$INICIAL;

```

```

81 1 PCALFABETA:PROCEDURE;
82 2 DO;
83 3 BUCLE3: B=INPUT(USARCONTROL) AND 01;
84 3 IF B=0 THEN GOTO BUCLE3;
86 3 OUTPUT(USARDATOS)='%' ;

87 3 DO I=1 TO 40;
88 4 CALL TIME(200);
89 4 END;
90 3 BUCLE4: B=INPUT(USARCONTROL) AND 02;
91 3 IF B=0 THEN GOTO BUCLE3;
93 3 ED=INPUT(USARDATOS);
94 3 IF ED<>'%' THEN GOTO BUCLE3;
96 3 DO I=0 TO 2; /* Recibo del PC a
- Ifa y beta e INOMINAL */
97 4 BUCLE2:B=INPUT(USARCONTROL) AND 01H;
98 4 IF B=0 THEN GOTO BUCLE2;
100 4 OUTPUT(USARDATOS)='&';
101 4 BUCLE: B=INPUT(USARCONTROL) AND 02H;
102 4 IF B=02 THEN GOTO ENTRADA;
104 4 GOTO BUCLE;
105 4 ENTRADA: IF I=0 THEN ALFA=INPUT(USARDATOS);
107 4 IF I=1 THEN BETA=INPUT(USARDATOS);
109 4 IF I=2 THEN INOMINAL=INPUT(USARDATOS);
111 4 END;
112 3 LAZO: B=INPUT(USARCONTROL) AND 01H; /*
- * Mando confirmacion al PC */
113 3 IF B=0 THEN GOTO LAZO; /* que lo
- s datos han sido */
115 3 OUTPUT(USARDATOS)='&'; /* recibi
- dos. */
116 3 PCAB10: RETURN;
117 3 END;
118 2 END PCALFABETA;

/* Fase de inicializacion del sistema */

119 1 COMIEN: DO;
120 2 OUTPUT(TIMERH$RAM1)=40H; /*Programacio
- n del klok del convertidor*/
121 2 OUTPUT(TIMERL$RAM1)=03CH; /* a 51 kHz
- aproximadamente */
122 2 OUTPUT(CONTROL$RAM1)=0C0H; /*Program
- acion de la USART en asincrono*/
123 2 OUTPUT(USARCONTROL)=0CFH; /*a 1200 bau
- dios, sin paridad y con dos*/
124 2 OUTPUT(TIMERH$RAM2)=40H; /*bits de STO

```

```

- P*/
125 2          OUTPUT(TIMERL$RAM2)=26H;
126 2          OUTPUT(CONTROL$RAM2)=0C0H;
127 2          OUTPUT(USARCONTROL)=07H;
128 2          OUTPUT(USARCONTROL)=37H;
129 2          OUTPUT(CONTROL$RAM1)=02H; /*PORT B p
- ara activar LED*/
130 2          OUTPUT(CONTROL$RAM2)=02H;
131 2          OUTPUT(PORTB$RAM1)=00H;
132 2  M01:     DO I=0 TO 2;
133 3           P(I)=0;
134 3           FLG(I)=0;
135 3           COR=0;
136 3           END;

/* Compruebo si el PC esta conectado */
137 2  B=INPUT(PORTA$RAM1) AND 01H;
138 2  IF B=01H THEN DO;
140 3      ALFA=10;
141 3      BETA=10;
142 3      INOMINAL=212;
143 3      GOTO CONEX1;
144 3      END;

145 2          CALL PCALFABETA;

146 2  CONEX1: CALL S$MASK(01DH); /* Programo masc
- ara de interrupciones 6.5 */
147 2  CALL TOMA$INICIAL; /* Hago una toma in
- icial de datos */

148 2  PRINCIPIO: OUTPUT(PORTB$RAM2)=LINDESCONECTADAS
- ; /*Actualizo neles*/
149 2          COR=0;
150 2  PRIN2:    B=INPUT(PORTA$RAM1) AND 01H;
151 2  PRIN3:    IF B=0 THEN DO;
153 3  PRIN4:    CALL TRANSMISION$AL$PC;
154 3          IF B='#' THEN CALL PCALFABETA;
156 3          END;
157 2          W=0;
158 2          DO I=0 TO 2;
159 3          IF IG(I)>=229 THEN
160 3              DO;
161 4  M04:      IF P(I)=0 THEN P(I)=128;
163 4  M05:      FLG(I)=1;
164 4          W=1;
165 4          END;
166 3          ELSE DO;

```

```

167 4          FLG(I)=0;
168 4          END;
169 3          END;
170 2      MO2:  IF W=1 THEN CALL RUTTESTEO(.P);

172 2      MO3:  DO I=0 TO 2;
173 3          IF IG(I)>= 195 THEN
174 3          DO;
175 4              P(I)=128 +(IG(I)-195)/3; /*Aproximaciones aceptables*/
176 4          END;
177 3          ELSE DO;
178 4              P(I)=0;
179 4          END;
180 3      MO10:  END;
181 2          GOTO PRINCIPIO;
182 2      MO11:  END;
183 1      END ESTABILIDAD;

```

MODULE INFORMATION:

```

CODE AREA SIZE      = 030DH      781D
VARIABLE AREA SIZE = 0019H      25D
MAXIMUM STACK SIZE = 0004H      4D
194 LINES READ
0 PROGRAM ERROR(S)

```

END OF PL/M-80 COMPILATION

ISIS-II PL/M-80 V3.1 COMPILATION OF MODULE MODTESTEO
 OBJECT MODULE PLACED IN RUTTE3.OBJ
 COMPILER INVOKED BY: PLM80 RUTTE3.PLM WORKFILES(:F0:,:F0:)
 -DEBUG PAGELength(54) PAGEWIDTH(60)

```

          /* Rutina de testeo durante un tiempo
-      T que depende
          del valor de P(I) */
1      MOD$TESTEO:DO;
2      1      DECLARE T(3) ADDRESS PUBLIC;

3      1      DECLARE IG(3) BYTE EXTERNAL;
4      1      DECLARE IL(5) BYTE EXTERNAL;
5      1      DECLARE FLG(3) BYTE EXTERNAL;
6      1      DECLARE (ALFA,BETA,INOMINAL) BYTE EXTERNAL
-      ;

7      1      DESCARGA:PROCEDURE EXTERNAL;
8      2      END DESCARGA;

9      1      TRANSMISION$AL$PC:PROCEDURE EXTERNAL;
10     2      END TRANSMISION$AL$PC;

11     1      RUTTESTEO:PROCEDURE (DIRPI) PUBLIC;
12     2      DECLARE (DIRPI) ADDRESS;
13     2      DECLARE (A,K,F,J,CONTADOR) BYTE;
14     2      DECLARE IN LITERALLY '212';
15     2      DECLARE PORTA$RAM1 LITERALLY '41H';
16     2      DECLARE P BASED DIRPI (3) BYTE;
17     2      RU1: DO;
18     3      CONTADOR=0; /*Contador para transmision
-      n al PC */
19     3      DO J=0 TO 2;
20     4      IF FLG(J)=1 THEN DO;
22     5      IF P(J)>138 THEN P(J)=138;
24     5      T(J)=1471-((1470*(P(J)-128))/10); /*Se c
-      alcula un T al que este mal.Este T se ira dec
-      rementando*/
25     5      RU3: END;
26     4      ELSE T(J)=1500; /*Se inicializan l
-      os otros T's */
27     4      END;
28     3      A=00H;
29     3      BASE: DO J=0 TO 2;
30     4      IF FLG(J)=0 THEN
31     4      DO;
32     5      IF IG(J)>=229 THEN
33     5      DO;
34     6      IF P(J)=0 THEN P(J)=128;

```

```

36 6          P(J)=P(J)+(IG(J)-195)/3;
37 6          IF P(J)>138 THEN P(J)=138;
39 6          T(J)=1471-(1470*(P(J)-128))/10;
40 6          FLG(J)=1;
41 6          END;
42 5          ELSE DO;
43 6          IF IG(J)>=195 THEN P(J)=128+(I
- G(J)-195)/3;
45 6          ELSE P(J)=0;
46 6          END;
47 5          END;
48 4          END;
49 3          F=0;
50 3          DO J=0 TO 2;
51 4              IF FLG(J)=1 THEN DO;
53 5                  IF T(J)=0 AND IG(J)>=229 THEN F=1;
55 5                  END;
56 4              END;
57 3          RU2:  IF F=1 THEN DO;
59 4              DO J=0 TO 2;
60 5                  FLG(J)=0;
61 5                  END;
62 4                  CALL DESCARGA;
63 4                  GOTO RU10;
64 4              END;
65 3              A=0;
66 3              DO J=0 TO 2;
67 4                  IF FLG(J)=1 THEN DO;
69 5                      IF T(J)=0 AND IG(J)<229 THEN FLG(J)=0;
71 5                      ELSE A=1;
72 5                  END;
73 4              END;
-          /*Realizar una transmision al PC cada 20 v
-          eces que se repita el bucle*/
74 3          IF CONTADOR=20 THEN DO;
76 4              CONTADOR=0;
77 4              K=INPUT(PORTA$RAM1) AN
- D 01H;
78 4              IF K=0 THEN CALL TRANSMISION$A
- L$PC;
80 4              END;
81 3              CONTADOR=CONTADOR+1;
82 3              IF A=1 THEN GOTO BASE;
84 3          RU10:  RETURN;
85 3          END;
86 2          END RUTTESTEO;
87 1          END MOD$TESTEO;

```

MODULE INFORMATION:
PL/M-80 COMPILER

PAGE 3

CODE AREA SIZE	= 02C0H	704D
VARIABLE AREA SIZE	= 000DH	13D
MAXIMUM STACK SIZE	= 0006H	6D
B1 LINES READ		
0 PROGRAM ERROR(S)		

END OF PL/M-80 COMPILATION

ISIS-II PL/M-80 V3.1 COMPILATION OF MODULE MODDESCARGA
 OBJECT MODULE PLACED IN DESCA3.OBJ
 COMPILER INVOKED BY: PLMB0 DESCA3.PLM WORKFILES(:F0:,:F0:)
 -DEBUG PAGELENGTH(54) PAGEWIDTH(60)

```

      /* Esta rutina tiene por objeto realizar l
- a descarga de lineas
      con el siguiente criterio: Calculo la line
- a o conjunto de lineas
      mas cercana al 10% de la potencia nominal
- de los generadores*/
1      MOD$DESCARGA:DO;
2      1      TRANSMISION$AL$PC:PROCEDURE EXTERNAL;
3      2      END TRANSMISION$AL$PC;

4      1      DECLARE IG(3) BYTE EXTERNAL;
5      1      DECLARE IL(5) BYTE EXTERNAL;
6      1      DECLARE (LINDESCONECTADAS,ALFA,BETA,INOMIN
- AL,COR) BYTE EXTERNAL;

7      1      DESCARGA:PROCEDURE PUBLIC;
8      2      DECLARE (D,CALCULO) ADDRESS;
9      2      DECLARE K BYTE;
10     2      DECLARE R(5) BYTE;
11     2      DECLARE (I,J,X,DESCONECTO,INTERMEDIA,
- INTER,DES) BYTE;

12     2      DECLARE PORTA$RAM1 LITERALLY '41H';
13     2      DECLARE PORTB$RAM1 LITERALLY '42H';
14     2      DECLARE PORTB$RAM2 LITERALLY '32H';
15     2      DECLARE CTE LITERALLY '45';
16     2      DECLARE IN$LINEA LITERALLY '153';
17     2      DECLARE R(5) BYTE;

      /* Ordeno las lineas de menor a mayor u qu
- ando en R el numero de
      de cada linea */
18     2      DECA: DO;
19     3      DO I=0 TO 4; /*Inicializo R(I) con
- su valor de origen*/
20     4      R(I)=IL(I);
21     4      GUARDALINEAS(I)=I;
22     4      END;
23     3      DE1: DO I=0 TO 3; /*Ordenamiento*/
24     4      DO X=I+1 TO 4;
25     5      IF R(X)<R(I) THEN
26     5      DO;
27     6      INTER=GUARDALINEAS(X);
28     6      INTERMEDIA=R(X);

```

```

29 6          GUARDALINEAS(X)=GUARDALINEAS(I);
30 6          R(X)=R(I);
31 6          GUARDALINEAS(I)=INTER;
32 6          R(I)=INTERMEDIA;
33 6          END;
34 5          END;
35 4          END;
-          /*Con los valores de ALFA y BETA recibidos e
-          establezco una forma de hallar
-          el 10% de la corriente nominal de los gene
-          radores*/
36 3          DE2:      D=(10+ALFA+BETA)*IN$LINEA;
-
-          /* Establezco una forma de eleccion de las l
-          inas a desconectar*/
37 3          DE3:      DO I=0 TO 4;
38 4              IF R(I)<45 THEN R(I)=45;
40 4              END;
41 3          DE4:      X=0;
42 3              CALCULO=(R(X)-CTE)*100;
43 3              DO WHILE CALCULO<D;
44 4                  X=X+1;
45 4                  CALCULO=(R(X)-CTE)*100;
46 4              END;
47 3          DE5:      IF X>4 THEN X=4;
49 3          DE6:      DO CASE GUARDALINEAS(X);          /*GUARDALI
-          NEAS(X) contiene un numero comprendido entre
-          0 y 4*/
50 4              DESCONECTO=11101111B;
51 4              DESCONECTO=11110111B;
52 4              DESCONECTO=11111011B;
53 4              DESCONECTO=11111101B;
54 4              DESCONECTO=11111110B;
55 4              END;
56 3              IF IL(GUARDALINEAS(X))<=45 THEN DO;
58 4                  OUTPUT(PORTB$RAM1)=03H;
59 4                  COR=01H;
60 4                  END;
61 3          DE7:      OUTPUT(PORTB$RAM2)=LINDESCONECTA
-          DAS AND DESCONECTO;
62 3              DO I=1 TO 4;
63 4                  CALL TIME(250);
64 4              END;
65 3              IF COR=01H THEN GOTO DE11;
67 3          DE8:      CALL TRANSMISION$AL$PC;
68 3              DO I=1 TO 14;          /*Retardo de 350 m
-          seg + tiempo de */

```

```

69  4          CALL TIME(250);      /*ejecucion de pro
-   programa ~ 0.5 seg */
70  4          END;

71  3          DE9:          DES=00;
72  3          DO I=0 TO 2;
73  4          IF IG(I)>=229 THEN DES=01; /*Despues d
-   e 0.5 seg de desconectada la linea */
75  4          END;          /*no se han
-   recuperado los generadores*/
76  3          K=INPUT(PORTA$RAM1) AND 01H;
77  3          DE10:         IF DES=1 THEN
78  3          DE12:         DO;
79  4          DE13:         IF K=0 THEN CALL TRANSMISION$AL$PC
-   ;
81  4          DE14:         GOTO DECA;
82  4          DE15:         END;
83  3          DE11:         IF K=0 THEN CALL TRANSMISION$AL$PC
-   ;
85  3          DE16:         RETURN;
86  3          END;

87  2          END DESCARGA;
88  1          END MOD$DESCARGA;

```

MODULE INFORMATION:

```

CODE AREA SIZE      = 0273H      627D
VARIABLE AREA SIZE = 0016H      22D
MAXIMUM STACK SIZE = 0002H      2D
101 LINES READ
0 PROGRAM ERROR(S)

```

END OF PL/M-80 COMPILATION

ISIS-II PL/M-80 V3.1 COMPILATION OF MODULE TOMADEDATOS
 OBJECT MODULE PLACED IN TDATA3.OBJ

COMPILER INVOKED BY: PLM80 TDATA3.PLM WORKFILES(:F0:,:F0:)
 -DEBUG PAGELength(54) PAGEWIDTH(60)

```

1          TOMA$DE$DATOS:DO;
2      1          DECLARE IG(3) BYTE EXTERNAL;
3      1          DECLARE IL(5) BYTE EXTERNAL;
4      1          DECLARE (INIC,LINDESCONECTADAS,CANALDA
-      C) BYTE EXTERNAL;
5      1          DECLARE FLG(3) BYTE EXTERNAL;
6      1          DECLARE T(3) ADDRESS EXTERNAL;
7      1          TDATOS:PROCEDURE PUBLIC;
8      2          DECLARE PORTA$RAM2 LITERALLY '31H';
9      2          DECLARE PORTB$RAM2 LITERALLY '32H';
10     2          DECLARE (B,I) BYTE;
11     2          DECLARE ADC(8) BYTE AT (2000H);
12     2      T1:      DO;
13     3          DO I=0 TO 2;
14     4          IF FLG(I)=1 THEN DO;
16     5          T(I)=T(I)-1;
17     5          END;
18     4          END;
19     3      T2:      IF CANALDAC<3 THEN DO;
21     4          IG(CANALDAC)=ADC(CANALDAC);
22     4          END;
23     3          ELSE DO;
24     4          IL(CANALDAC-3)=ADC(CANALDAC);
25     4          END;

26     3          CANALDAC=CANALDAC+1;
27     3      T3:      IF CANALDAC=08H THEN DO;
29     4          LINDESCONECTADAS=INPUT(PORTA$RAM2)
-      ;

30     4          CANALDAC=0;
31     4          INIC=0;
32     4          END;
33     3          ADC(CANALDAC)=0;
34     3      T10:     RETURN;
35     3          END;
36     2          END TDATOS;
37     1          END TOMA$DE$DATOS;

```

MODULE INFORMATION:

```

CODE AREA SIZE      = 0097H      151D
, VARIABLE AREA SIZE = 0002H      2D

```

MAXIMUM STACK SIZE = 0002H 2D
35 LINES READ
0 PROGRAM ERROR(S)

END OF PL/M-80 COMPILATION

LOC	OBJ		LINE	SOURCE STATEMENT
			1	EXTRN TDATOS, COMIEN
0000			2	ORG 0
0000	C50000	E	3	JMP COMIEN
0034			4	ORG 34H
0034	C50000	C	5	JMP RI1
			6	CSEG
0000	F3		7	RI1: DI
0001	F5		8	PUSH PSW
0002	E5		9	PUSH H
0003	D5		10	PUSH D
0004	C5		11	PUSH E
0005	CD0000	E	12	CALL TDATOS
0008	C1		13	POP B
0009	D1		14	POP D
000A	E1		15	POP H
000B	F1		16	POP PSW
000C	FB		17	EI
000D	C9		18	RI10: RET
			19	END

PUBLIC SYMBOLS

EXTERNAL SYMBOLS

COMIEN E 0000 TDATOS E 0000

USER SYMBOLS

COMIEN E 0000 RI1 C 0000 RI10 C 000D TDATOS E 0000

ASSEMBLY COMPLETE, NO ERRORS

P R O G R A M A D E L O R D E N A D O R

```

10 COLOR 7,0
20 CLS
30 SCREEN 0,0,0,0:KEY OFF
40 T$="a"
50 DIM A(20)
60 A$=""
70 CLS:PRINT "Entra parametros de los generadores "
80 PRINT "Potencia nominal : "
90 FOR I=1 TO 3
100 PRINT"GENERADOR ";I;";";
110 INPUT X(I)
120 NEXT I
130 CLOSE R1
140 OPEN "com1:1200,n,8,2" AS R1
150 CLS:IF ASC(T$)=5 THEN GOTO 160
160 REM calculo y mando los valores de alfa y beta
170 ALFA =(X(2)/X(1))*10
180 BETA=(X(3)/X(1))*10
190 INOMINAL=212 :MI$="":MA$="":ME$=""
200 IF EOF(1) THEN GOTO 200
210 MI$=MI$+INPUT$(LOC(1),R1)
220 F$=INPUT$(1,R1)
230 IF F$<>"%" THEN GOTO 200
240 PRINT R1,"% "
250 IF EOF(1) THEN GOTO 250
260 F$=INPUT$(1,R1)
270 IF F$<>"&" THEN GOTO 250
280 PRINT R1,CHR$(ALFA)
290 IF EOF(1) THEN GOTO 290
300 F$=INPUT$(1,R1)
310 IF F$<>"&" THEN GOTO 290
320 PRINT R1,CHR$(BETA)
330 IF EOF(1) THEN GOTO 330
340 F$=INPUT$(1,R1)
350 IF F$<>"&" THEN GOTO 330

```

```

360 PRINT R1,CHR$(INOMINAL)
370 IF EOF(1) THEN 370
380 V$=INPUT$(1,R1)
390 IF V$<>"&" THEN PRINT "FALLO EN LA TRANSMISION DE ALFA Y BETA":STOP
400 SWITCH=1
410 CLS
420 GOSUB 1400
430 A$=""
440 PRINT R1,"$"
450 FOR I=1 TO 100
460 A=0
470 NEXT I
480 T$=INKEY$:IF LEN(T$)<>0 THEN GOSUB 1830
490 IF EOF(1) THEN GOTO 440
500 A$=INPUT$(1,R1)
510 IF A$<>"$" THEN GOTO 440
520 A$=""
530 A$=A$+INPUT$(1,R1)
540 IF LEN(A$)=9 THEN 560
550 GOTO 530
560 IF SWITCH=0 THEN 430
570 FOR I=1 TO 9
580 A(I)=ASC(MID$(A$,I,1))
590 NEXT I
600 FOR I=1 TO 3
610 A(I)=INT((A(I)-45)*56.28743)
620 NEXT I
630 FOR I=4 TO 8
640 A(I)=INT((A(I)-45)*61.30437 )
650 NEXT I
660 B$=""
670 C=A(9)
680 I=0
690 Q=FIX(C/2)
700 IF C/2=FIX(C/2) THEN B$="0"+B$:GOTO 720

```

```

710 B$="1"+B$
720 I=I+1
730 IF C<2 THEN 760
740 C=0
750 GOTO 690
760 IF I<8 THEN B$="0"+B$
770 I=I+1:IF I>8 THEN 790
780 GOTO 760
790 FOR I=1 TO 8
800 B$(I)=MID$(B$,I,1)
810 IF B$(I)="1" THEN C$(I)="ON" ELSE C$(I)="OFF"
820 NEXT I
830 COLOR 0,7
840 FOR I=1 TO 3
850 LOCATE 4+2*I,36:PRINT "          "
860 X$(I)=STR$(X(I))
870 IF X(I)>9999 THEN LOCATE 4+2*I,37:PRINT LEFT$(X$(I),3);". ";RIGHT$(X$(I),3):G
OTO 920
880 IF X(I)> 999 THEN LOCATE 4+2*I,38:PRINT LEFT$(X$(I),2);". ";RIGHT$(X$(I),3)
890 IF LEN(X$(I))=3 THEN LOCATE 4+2*I,40:PRINT X(I)
900 IF LEN(X$(I))=2 THEN LOCATE 4+2*I,41:PRINT X(I)
910 IF LEN(X$(I))=1 THEN LOCATE 4+2*I,42:PRINT X(I)
920 NEXT I
930 R=0:L=0
940 FOR I=1 TO 3
950 R=R+A(I)
960 NEXT I
970 FOR I=4 TO 8
980 L=L+A(I)
990 NEXT I
1000 IF ABS(L-R)>=.02*R THEN SUMA=1
1010 LOCATE 2,59:PRINT "          ":LOCATE 2,69:PRINT "          "
1020 IF SUMA=1 THEN COLOR 16,7:LOCATE 2,55:PRINT "EG=";R;" e IL=";L :COLOR 7,0
1030 IF SUMA=0 THEN LOCATE 2,55:PRINT "EG=";R;" e IL=";L
1040 SUMA=0
1050 U=0

```

```
1060 FOR I=1 TO 8
1070 IF A(I)<0 THEN U=U+1:E(U)=I
1080 NEXT I
1090 IF U >0 THEN A(10)=A(10) OR 4
1100 FOR I=1 TO 3
1110 LOCATE 4+2*I,57:PRINT "      "
1120 IF C$(I)="ON" THEN COLOR 0,7 ELSE COLOR 16,7
1130 LOCATE 4+2*I,58:PRINT C$(I)
1140 NEXT I
1150 FOR I=4 TO 8
1160 LOCATE 6+2*I,57:PRINT "      "
1170 IF C$(I)="ON" THEN COLOR 0,7 ELSE COLOR 16,7
1180 LOCATE 6+2*I,58:PRINT C$(I)
1190 NEXT I
1200 COLOR 0,7
1210 FOR I=4 TO 6
1220 LOCATE 2*I-2,17:PRINT "      "
1230 IF A(I-3)<X(I-3) THEN COLOR 0,7 ELSE COLOR 16,7:GOSUB 1360
1240 LOCATE 2*I-2,16:PRINT A(I-3)
1250 COLOR 0,7
1260 NEXT I
1270 COLOR 0,7
1280 FOR I=7 TO 10
1290 LOCATE 2*I,17:PRINT "      "
1300 LOCATE 2*I,16:PRINT A(I-3)
1310 NEXT I
1320 LOCATE 22,17:PRINT "      "
1330 LOCATE 22,16:PRINT A(8)
1340 COLOR 7,0
1350 GOTO 430
1360 LOCATE 2*I-2,16:PRINT A(I-3)
1370 BEEP:BEEP:BEEP
1380 RETURN 1260
1390 CLS
1400 COLOR 7,0
```

```

1410 LOCATE 1,1:PRINT CHR$(201);STRING$(26,CHR$(205));CHR$(203);STRING$(24,CHR$(
205));CHR$(203);STRING$(25,CHR$(205));CHR$(187)
1420 LOCATE 2,28:PRINT CHR$(186):LOCATE 2,53:PRINT CHR$(186)
1430 LOCATE 3,28:PRINT CHR$(200);STRING$(24,CHR$(205));CHR$(188)
1440 FOR A=2 TO 23:LOCATE A,1:PRINT CHR$(186):NEXT A
1450 FOR A=2 TO 23:LOCATE A,79:PRINT CHR$(186):NEXT A
1460 LOCATE 23,1:PRINT CHR$(200);STRING$(77,CHR$(205));CHR$(188)
1470 FOR A=15 TO 55 STEP 20
1480 LOCATE 5,A:PRINT CHR$(218);STRING$(8,CHR$(196));CHR$(191)
1490 LOCATE 6,A:PRINT CHR$(179):LOCATE 6,A+9:PRINT CHR$(179)
1500 LOCATE 7,A:PRINT CHR$(195);STRING$(8,CHR$(196));CHR$(180)
1510 LOCATE 8,A:PRINT CHR$(179):LOCATE 8,A+9:PRINT CHR$(179)
1520 LOCATE 9,A:PRINT CHR$(195);STRING$(8,CHR$(196));CHR$(180)
1530 LOCATE 10,A:PRINT CHR$(179):LOCATE 10,A+9:PRINT CHR$(179)
1540 LOCATE 11,A:PRINT CHR$(192);STRING$(8,CHR$(196));CHR$(217)
1550 NEXT A
1560 FOR A=15 TO 55 STEP 40
1570 LOCATE 13,A:PRINT CHR$(218);STRING$(8,CHR$(196));CHR$(191)
1580 LOCATE 14,A:PRINT CHR$(179):LOCATE 14,A+9:PRINT CHR$(179)
1590 LOCATE 15,A:PRINT CHR$(195);STRING$(8,CHR$(196));CHR$(180)
1600 LOCATE 16,A:PRINT CHR$(179):LOCATE 16,A+9:PRINT CHR$(179)
1610 LOCATE 17,A:PRINT CHR$(195);STRING$(8,CHR$(196));CHR$(180)
1620 LOCATE 18,A:PRINT CHR$(179):LOCATE 18,A+9:PRINT CHR$(179)
1630 LOCATE 19,A:PRINT CHR$(195);STRING$(8,CHR$(196));CHR$(180)
1640 LOCATE 20,A:PRINT CHR$(179):LOCATE 20,A+9:PRINT CHR$(179)
1650 LOCATE 21,A:PRINT CHR$(195);STRING$(8,CHR$(196));CHR$(180)
1660 LOCATE 22,A:PRINT CHR$(179):LOCATE 22,A+9:PRINT CHR$(179)
1670 LOCATE 23,A:PRINT STRING$(11,CHR$(205))
1680 NEXT A
1690 LOCATE 23,15:PRINT CHR$(193):LOCATE 23,24:PRINT CHR$(193)
1700 LOCATE 23,55:PRINT CHR$(193):LOCATE 23,64:PRINT CHR$(193)
1710 LOCATE 2,30:COLOR 16,7:PRINT "CONTROL DE ESTABILIDAD":COLOR 7,0
1720 LOCATE 4,15:PRINT "POTENCIA":LOCATE 4,33:PRINT "POT.NOMINAL":LOCATE 4,56:PR
INT "ESTADO"
1730 LOCATE 6,2:PRINT "Generador 1:":LOCATE 6,26:PRINT "KVA":LOCATE 6,46:PRINT "
KVA"
1740 LOCATE 8,2:PRINT "Generador 2:":LOCATE 8,26:PRINT "KVA":LOCATE 8,46:PRINT "
KVA"

```

```
1760 LOCATE 12,15:PRINT "POTENCIA":LOCATE 12,56:PRINT "ESTADO"  
1770 LOCATE 14,2:PRINT "Linea 1:      ":LOCATE 14,25:PRINT " KVA"  
1780 LOCATE 16,2:PRINT "Linea 2:      ":LOCATE 16,25:PRINT " KVA"  
1790 LOCATE 18,2:PRINT "Linea 3:      ":LOCATE 18,25:PRINT " KVA"  
1800 LOCATE 20,2:PRINT "Linea 4:      ":LOCATE 20,25:PRINT " KVA"  
1810 LOCATE 22,2:PRINT "Linea 5:      ":LOCATE 22,25:PRINT " KVA"  
1820 RETURN  
1830 IF ASC(T$)=21 THEN SWITCH=0:RETURN  
1840 IF ASC(T$)=5 THEN PRINT R1,"R":GOTO 70  
1850 SWITCH=1  
1860 RETURN
```


E S T U D I O D E A L G U N O S

C I R C U I T O S

L S I U T I L I Z A D O S

Se considera conveniente la exposición del funcionamiento de algunos circuitos integrados utilizados en el prototipo, por poder requerirse estos conocimientos a la hora de realizar la reparación de averías en el mismo.

Los circuitos integrados de tecnología LSI que se tratarán son aquellos cuyo funcionamiento resulta complicado o poco intuitivo cuando nos encontramos ante ellos por primera vez. Se intentará ser claros y breves en la medida de lo posible, pues lo que se pretende no es realizar un estudio detallado, sino dar una idea de su filosofía de funcionamiento y forma de manejo.

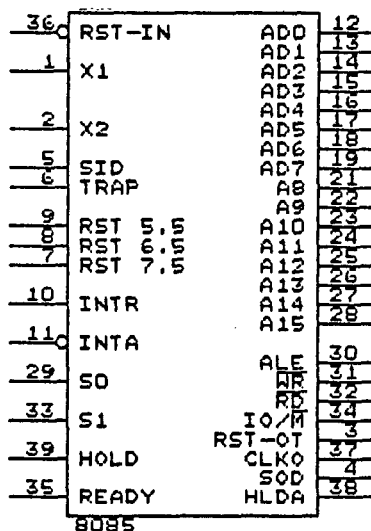
Se hará un estudio, por tanto, de los siguientes integrados:

- el microprocesador 8085.
- el C.I. 8155 (memoria RAM-puertos-timer).
- la USART 8251.
- el convertidor analógico-digital 0808.

EL MICROPROCESADOR 8085

El microprocesador 8085 es una versión posterior y mejorada del 8080 de Intel, que procede a su vez del 8008.

Se trata de un microprocesador de 8 bits, bastante versátil por el gran número de aplicaciones que tiene. Físicamente se compone de un único chip, de tecnología NMOS, en el que se integran unos 6200 transistores, dentro de una cápsula de 40 patillas dual-in-line, con un tamaño de 164 x 222 milésimas de pulgada. En la figura se muestra su configuración externa.



La tensión de alimentación es de 5 V (entre V_{cc} y masa o patilla V_{ss}). Necesita un cristal externo o red RC que le proporcione una señal de reloj adecuada, entre las patillas X1 y X2. El microprocesador trabajará a una frecuencia básica que es la mitad de esta frecuencia proporcionada por el cristal; así, si el cristal es de 6,25 MHz, la frecuencia interna de trabajo será de 3,125 MHz. Además, el 8085 posee una salida de reloj a esta frecuencia de trabajo, con el fin de poder sincronizar dispositivos externos.

Este microprocesador tiene capacidad para direccionar hasta 64K posiciones de memoria, con sus 16 patillas de direccionamiento. Ocho de los 16 bits necesarios, A8-A15, se obtienen directamente de las patillas triestado A8-A15; los restantes, A0-A7, se obtienen en las patillas bidireccionales triestado AD0-AD7, que además son las patillas destinadas a los datos. Es decir, estas ocho patillas están multiplexadas en el tiempo: en unos momentos contienen direcciones y en otros datos.

Es necesario, por tanto, un latch de direcciones de ocho bits que memorice la información de estas patillas cuando en ellas haya direcciones. La señal ALE del microprocesador se activa en este caso, y actúa de enable

para el latch de direcciones. Con el multiplexaje de direcciones/datos se consigue la reducción del número de patillas necesarias en la cápsula.

A la hora de realizar una referencia a memoria se transmite un byte de datos en las patillas de direcciones/datos; se pueden dar dos situaciones;

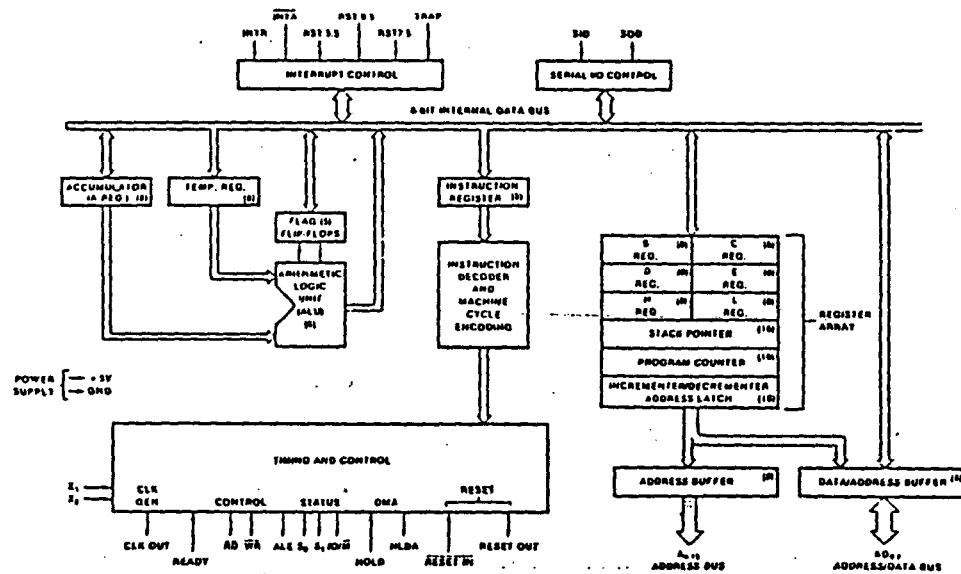
- una lectura de datos: el microprocesador activa la señal de lectura, RD, y un dato de 8 bits es transferido desde memoria al microprocesador.
- una escritura de datos: el microprocesador activa la señal WR de escritura y el dato es transferido desde el microprocesador a memoria.

También se nos permite direccionar hasta 256 puertos de entrada y 256 puertos de salida, utilizando las instrucciones de E/S proporcionadas por el microprocesador, y los 8 bits disponibles para el direccionamiento de puertos. La señal de control IO/M del 8085 indica si la operación de lectura/escritura va a ser realizada en un port o en una posición de memoria: un nivel alto en esta patilla selecciona los ports; un nivel bajo selecciona la memoria.

En cuanto a su arquitectura, el 8085 consta de una

serie de registros internos ilustrados en la figura:

- un contador de programa de 16 bits, PC.
- un puntero de pila de 16 bits, SP o stack pointer.
- seis registros de propósito general de 8 bits dispuestos en pares: BC, DE, HL.
- una pareja de registros temporales, WZ.
- una unidad lógico-aritmética, ALU.



Veamos un poco cómo trabajan. Cuando se hace un reset en el microprocesador (la patilla RESET IN se lleva a nivel bajo), el contador de programa se pone a 0; cuando RESET IN vuelve a 1, la unidad de control transfiere el contenido del PC al registro de direcciones, y así proporciona la dirección de la primera instrucción a ejecutar: la ejecución del programa empieza en la posición cero.

Las instrucciones pueden ser de hasta tres bytes, el primero de ellos siempre es el código de operación; los otros son datos o direcciones. El código de operación se carga en el registro de instrucciones, y al decodificarse se generan las microórdenes necesarias para la ejecución de la instrucción.

Los seis registros de propósito general del microprocesador pueden ser utilizados en parejas, como registros de 16 bits: HL, DE, BC; o bien individualmente como registros de 8 bits. El registro WZ no es direccionable por programa; se trata de un registro temporal utilizado por la unidad de control para la ejecución interna de instrucciones.

El registro SP se utiliza como puntero de la parte superior de la pila, localizada en memoria RAM externa.

La ALU es la encargada de las operaciones lógicas y aritméticas. Los operandos se almacenan en el acumulador, registro de 8 bits, y el registro temporal, también de 8 bits.

Los bits de estado del 8085 se guardan en el registro de 5 bits F, asociado con la ALU. Indican la paridad, acarreo auxiliar, acarreo del bit de mayor peso, resultado cero de una operación y signo.

El bus interno de datos es de 8 bits, por lo que se necesita el buffer latch para el bus de direcciones/datos multiplexado. El latch del bus de direcciones/datos aísla el bus de datos interno del microprocesador del bus de direcciones/datos externo del sistema, por su característica triestado.

Tiempos y secuenciamiento.-

Se llama ciclo de instrucción a la búsqueda y ejecución de una instrucción. Un ciclo se descompone en varias operaciones denominadas ciclos de máquina. En el 8085 existen siete ciclos de máquina distintos. Además, cada ciclo de máquina es dividido por el reloj del sistema en

un número de estados T, que es el periodo de dicho reloj.

Los ciclos de máquina del 8085 son los siguientes:

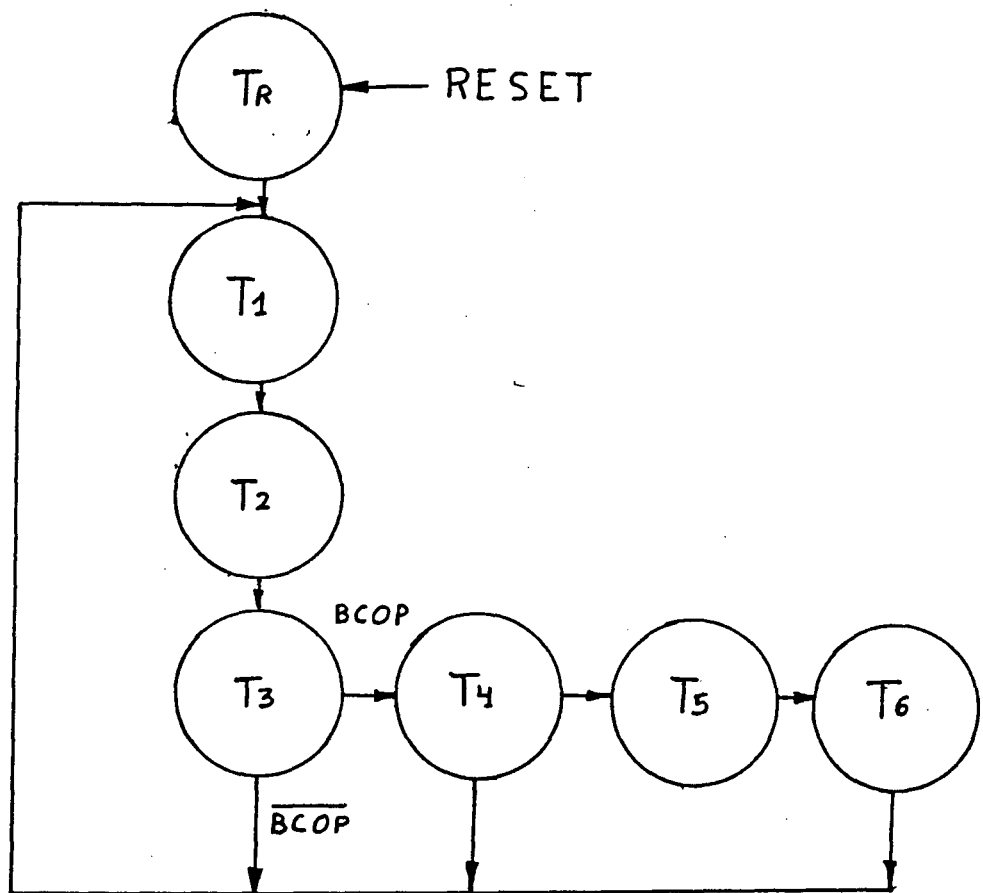
- búsqueda del código de operación.
- lectura de memoria.
- escritura de memoria.
- lectura de E/S.
- escritura de E/S.
- respuesta a interrupción.
- bus inactivo.

Cada uno de ellos comienza en un estado T1, y se van sucediendo T2, T3,...hasta acabar dicho ciclo de máquina. Después comienza el siguiente ciclo de máquina de la instrucción, otra vez a partir de T1.

Como máximo, un ciclo de máquina puede tener seis estados, desde T1 a T6, y como mínimo tres, desde T1 a T3. En total, una instrucción del 8085 puede constar de 4 a 18 estados; si consideramos una frecuencia interna de trabajo de 3,125 MHz (frecuencia para la cual un estado dura 320 ns), la ejecución de una instrucción puede durar entre 1,3 y 5,8 microsegundos.

El tiempo mínimo de estado para el 8085 es de 320 ns, y su tiempo máximo de 2000 ns.

Los ciclos de máquina de búsqueda de instrucción pueden tener entre 4 y 6 estados; los demás ciclos constan de 3 estados. En la figura aparece un diagrama donde se refleja la transición de estados comentada:



TRANSICIÓN DE ESTADOS

BCOP = búsqueda código operación

Se ha de procurar que los dispositivos externos al microprocesador sean tan rápidos como éste. Si no es así, el dispositivo externo deberá pedir al microprocesador que espere un cierto número de estados.

Esto se consigue colocando la entrada READY del microprocesador a cero; durante el estado T2 de la instrucción en curso, el microprocesador testea esa patilla, y si detecta el cero entrará automáticamente en un estado de espera, T_{wait}, en vez de entrar en T3. Cuando el dispositivo externo que solicitó la espera termine su trabajo, la patilla READY del microprocesador volverá a 1, y el microprocesador continuará su ciclo habitual.

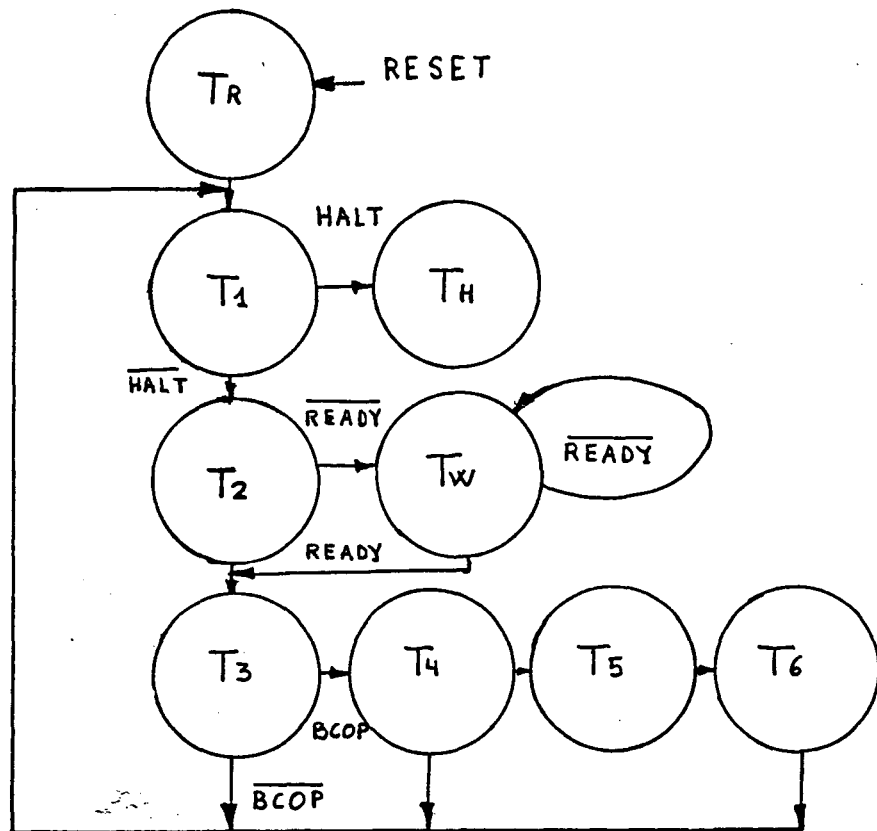
Otro estado posible en el 8085 es el de HALT o parada. La instrucción HALT se decodifica, como todas, durante el estado T4. Al reconocer esta instrucción un biestable se coloca a uno lógico.

Después del T4 el microprocesador vuelve a T1, y como consecuencia de ello se entrará en el estado de paro T_{HALT}, en lugar del estado T2. Con el reseteo del microprocesador se podrá salir de este estado.

Un reset del 8085 coloca todos sus registros a cero; el reset se propaga al resto del sistema a través de la

patilla RESET OUT.

Teniendo en cuenta lo dicho, un diagrama simplificado de la secuencia de estados más completo que el anterior es el siguiente:



TRANSICIÓN DE ESTADOS CON Twait y Thalt

Interrupciones.-

Como lo que se intenta no es un estudio exhaustivo del 8085, sino una visión del mismo para conocer su modo de funcionamiento, se explicarán sólo aquellas interrupciones utilizadas en el proyecto, y simplemente se nombrarán las demás.

Existen dos tipos de interrupciones: las enmascarables y las no enmascarables. Cuando se produce una no enmascarable, el microprocesador será siempre interrumpido. Sin embargo, si la interrupción es de tipo enmascarable, será interrumpido si la interrupción no ha sido enmascarada.

El 8085 posee cinco tipos de interrupciones: TRAP, RST 7.5, RST 6.5, RST 5.5, e INTR. La interrupción TRAP no es enmascarable, todas las demás sí lo son.

La TRAP se utiliza para casos catastróficos, como fallo en la alimentación. La INTR permite siete tipos diferentes, pudiendo existir siete dispositivos distintos que la activen, y pudiendo implementarse siete rutinas distintas para dar servicio a cada una. Este tipo de interrupción utiliza las patillas INTR e INTA: INTR para producir la interrupción desde el exterior; INTA para

indicar al dispositivo externo que ha sido reconocida la interrupción. Si se desea más información sobre estas interrupciones, recórrase a la bibliografía correspondiente.

Las interrupciones que realmente nos interesa son las restantes: RST 7.5, RST 6.5 Y RST 5.5. Como se dijo, estas interrupciones son enmascarables. El programa permitirá habilitarlas, y para ello se precisan dos operaciones:

- programación de las máscaras de interrupción.
- habilitación de las interrupciones.

El 8085 proporciona la instrucción SIM y EI para este propósito. La ejecución de SIM sirve para programar las máscaras, dependiendo del contenido del acumulador en el momento de su ejecución.

Los bits 0, 1 y 2 del acumulador ponen a 1 o a 0 el bit de máscara de la RST 7.5, RST 6.5 y RST 5.5 respectivamente.

El bit 3 es el control de puesta de máscaras; esto es, si el bit 3 contiene un 1, las máscaras serán programadas según se indique en los bits 0, 1 y 2. Si está a 0, la programación de máscaras permanecerá como estaba antes.

La habilitación de interrupciones se realiza mediante la instrucción EI del 8085. La ejecución de la misma provoca la puesta a uno de un biestable de habilitación de interrupciones, INTE. Por el contrario, la instrucción DI resetea dicho biestable, inhabilitando las interrupciones.

Una vez producida una interrupción, el flip-flop INTE es puesto a cero (se inhiben automáticamente las interrupciones). Si queremos que el microprocesador vuelva a aceptar una segunda interrupción habrá que ejecutar de nuevo una instrucción EI.

El 8085 dispone de otra instrucción, RIM (lectura de máscara de interrupciones), que da información sobre el estado de las máscaras de interrupción. Cuando se ejecuta una instrucción RIM el acumulador se carga con un byte cuyo significado es el siguiente:

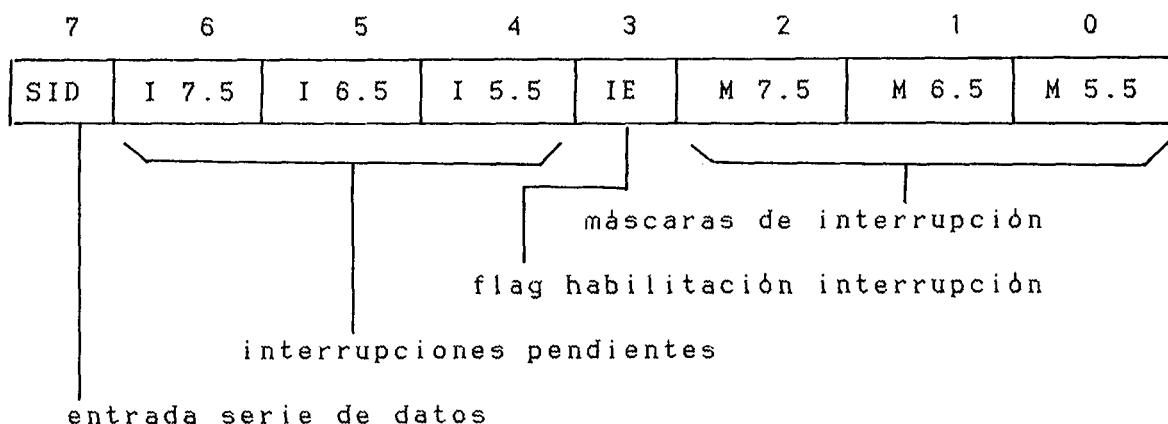
- el bit 0, 1 y 2 indican el estado de la máscara de las interrupciones RST 5.5, RST 6.5 Y RST 7.5 respectivamente. Si hay un uno significa que la máscara correspondiente está colocada y viceversa.

- el bit 3 muestra el estado de la habilitación de interrupciones: un 1 indica que están habilitadas las interrupciones enmascarables, y un 0 que están deshabilitadas.

- el bit 4, 5 y 6 indican con un uno la existencia de una interrupción RST 5.5, RST 6.5 y RST 7.5 pendientes de ser atendidas, respectivamente.

- el bit 7 es el correspondiente a la entrada de datos serie por la patilla SID del microprocesador.

El esquema siguiente ilustra esta distribución:



CONTENIDO DEL ACUMULADOR DESPUES DE UNA INSTRUCCION RIM

Las entradas de interrupción son comprobadas por el 8085 durante el flanco de reloj del penúltimo al último estado de cada ciclo de instrucción, y durante cada ciclo de reloj si el microprocesador está en estado HALT.

La entrada RST 7.5 es sensible a flancos positivos. Este flanco es memorizado en un flip-flop, que volverá a

cero cuando la interrupción sea atendida. Las RST 6.5 y RST 5.5 son sensibles a nivel 1. Este uno se debe mantener hasta ser reconocida. En la RST 7.5, sin embargo, la entrada de interrupción puede volver a cero, pues la petición queda memorizada en el flip-flop.

Un ciclo de máquina más completo que los mostrados en esquemas anteriores es el de la figura de la página siguiente, donde se incluye la lógica de interrupciones.

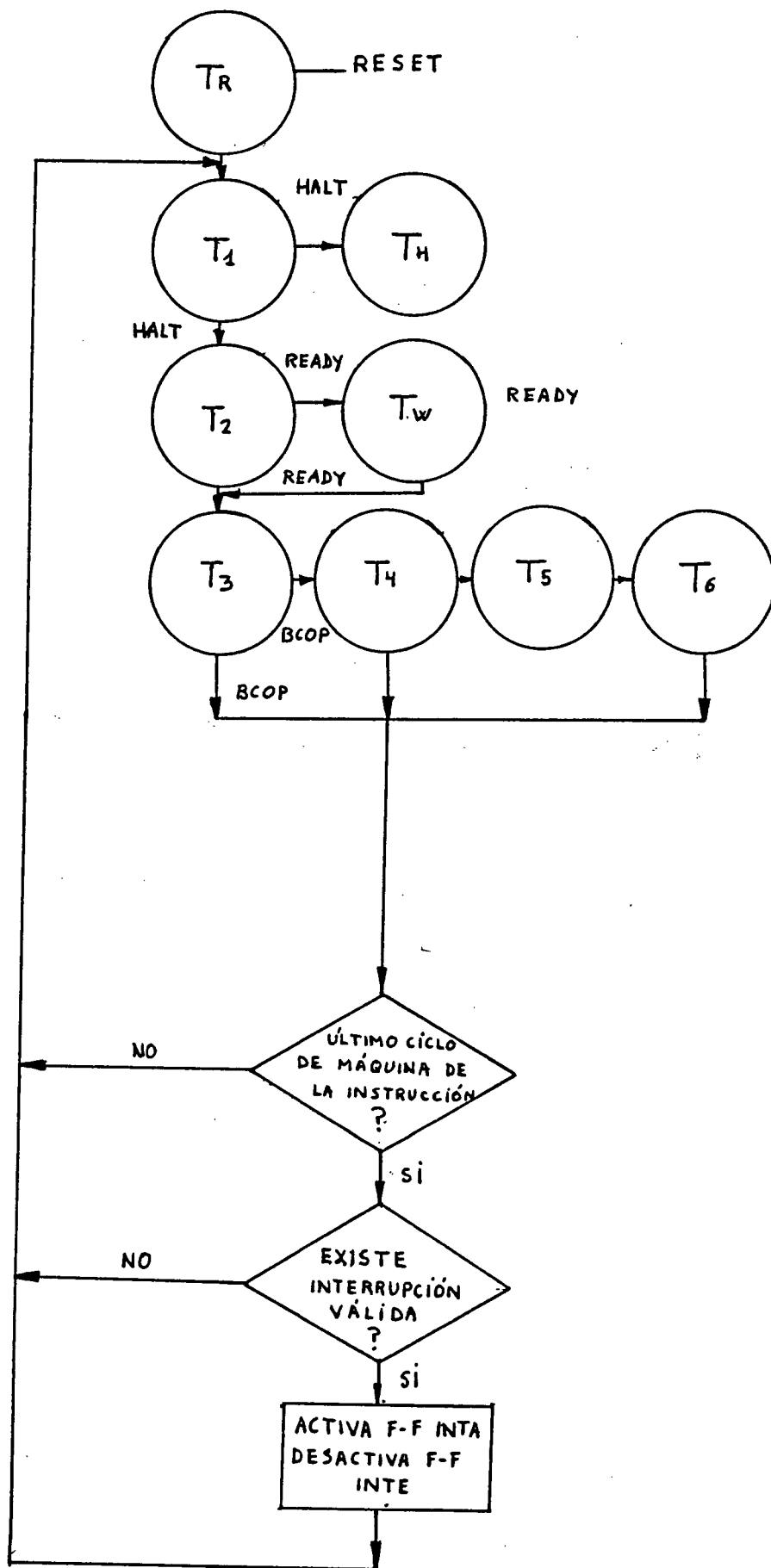
Cuando se reciben las interrupciones, el microprocesador transfiere el control a una dirección predeterminada. Para las interrupciones que estamos estudiando estas direcciones son:

RST 7.5 ----- 3C H

RST 6.5 ----- 34 H

RST 5.5 ----- 2C H

En caso de producirse estas tres interrupciones al mismo tiempo, el orden de prioridad de ejecución será el siguiente: la instrucción de mayor prioridad es la RST 7.5; la RST 6.5 le sigue; la última en ser atendida sería la RST 5.5.



Por último, los pasos que se siguen cuando se produce una interrupción son los siguientes:

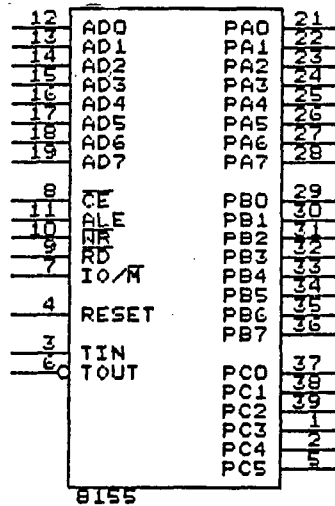
- 1) Se determina la instrucción que se está ejecutando en el momento en que se produce la interrupción.
- 2) Se ejecuta un ciclo de máquina de interrupción, que produce el almacenamiento del PC y la transferencia del control a la posición de memoria donde se encuentra la subrutina de interrupción.
- 3) En la subrutina de interrupción se guardan los registros de la CPU.
- 4) Se identifica el dispositivo que ha pedido la interrupción, si más de uno está asociado con esa interrupción.
- 5) Se ejecuta la subrutina que da servicio al dispositivo de E/S que ha provocado la interrupción.
- 6) Una vez acabada la subrutina, y antes de retornar, se restauran los registros del microprocesador que hablan sido guardados en memoria.
- 7) Se retorna al punto donde se había producido la

interrupción, transfiriéndose al PC la posición de memoria correspondiente, que había sido salvada al producirse la interrupción, y se continúa ejecutando el programa.

EL CIRCUITO INTEGRADO 8155

El 8155 es un chip empaquetado en una cápsula de 40 patillas. Contiene 256 bytes de memoria de lectura-escritura, dos puertos de E/S de 8 bits y uno de 6 bits, además de un temporizador programable de 14 bits. Su bus de datos y direcciones está multiplexado.

La parte I/O, como se dijo, consiste en tres puertos de propósito general. Uno de los tres puertos puede programarse para ser bits de status, mientras los otros dos puertos funcionan en modo "handshake".



Función de las patillas del 8155.-

- RESET: El impulso suministrado por el 8085 a través de su patilla RESET OUT provoca un reseteo del 8155, inicializando los tres puertos en el modo de entrada.
- AD0-AD7: Son las líneas de datos, multiplexadas con las ocho líneas menos significativas del bus de direcciones.
- CE: Se trata del habilitador del chip.
- RD: Un cero en esta línea, y un uno en CE, permite la lectura de un dato colocado en el bus de datos, procedente de IO o de RAM.
- WR: Un cero en ella, y un uno en CE originará que el dato de las líneas AD sea escrito en la RAM o en las puertas I/O, dependiendo de la polaridad de la patilla IO/M.
- ALE: Permiso del latch de direcciones: latchea las direcciones de las líneas AD0-AD7, y el estado de CE e IO/M en el chip en el flanco de bajada.
- IO/M: Selecciona entre puertos y memoria RAM.
- PA0-PA7: Son ocho patas I/O correspondientes a las del puerto A del integrado. Se puede

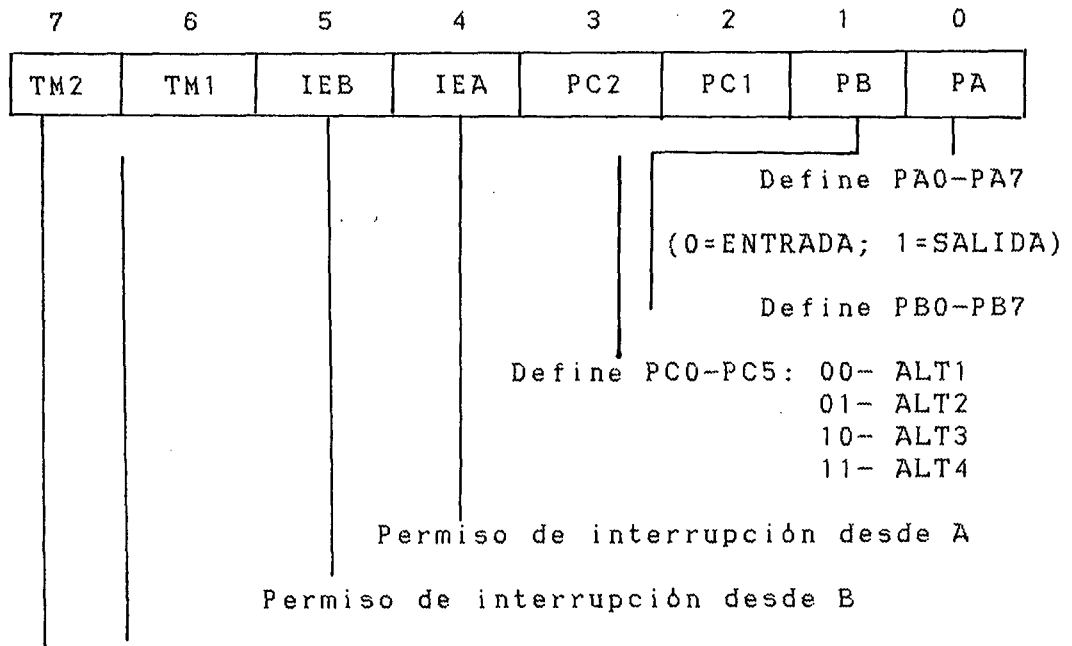
programar como puerto de entrada o de salida.

- PB0-PB7: Idem del puerto B.
- PC0-PC5: Son seis patillas que pueden funcionar tanto como puerto de entrada, de salida, o como señales de control para los puertos A y B. La programación se realiza a través del registro de comando/status.
- TIMER IN: Es la entrada de reloj para el contador.
- TIMER OUT: La salida del timer del chip 8155.
- VCC y VSS: Son las patillas de alimentación (5 V y masa).

Programación del registro de comando/status.-

El registro de comando consiste en ocho latches, uno para cada bit. Cuatro de ellos (del 0 al 3) definen el modo de los puertos, dos bits (el 4 y 5) permiten o impiden la interrupción desde la puerta C cuando ésta actúa como puerta de control, y los dos últimos bits (el 6 y 7) son para el timer. En el esquema siguiente se muestra esta distribución.

DEFINICION DEL REGISTRO DE COMANDO/STATUS



Comando del timer: 00- NOP- no afecta al funcionamiento del contador.

01- STOP- NOP (no operar) si el timer no ha comenzado. Para el contaje si el timer está corriendo.

10- STOP después del fin de contaje (TC). Para inmediatamente después que se alcance el actual TC. No opera si el timer no ha empezado.

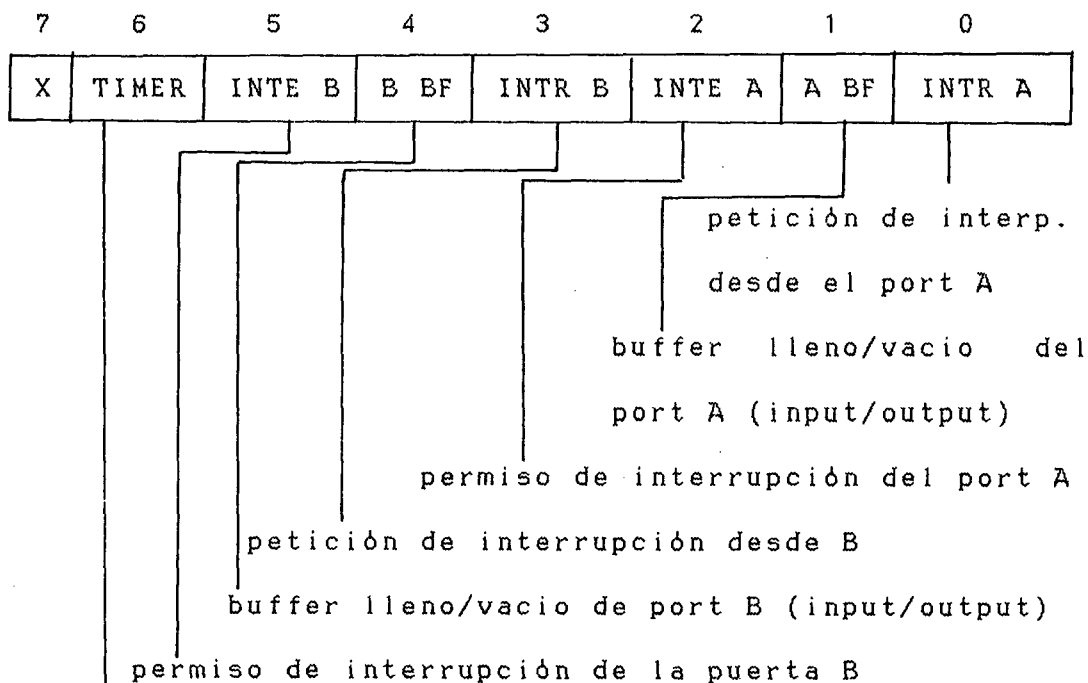
11- ARRANQUE- carga modo y anchura del contador y comienza inmediatamente después de cargar (si el timer no está actualmente en marcha). Inicia nuevo modo y anchura del contador después que se alcance el TC.

Lectura del registro de comando/status.-

El registro de status consiste en siete latches, uno para cada bit; seis (0 a 5) para el estado de las puertas, y uno (6) para el estado del timer.

El estado del timer y de la parte I/O puede sondearse leyendo el registro C/S (dirección X X X X X 0 0 0). En el esquema se muestra el formato de la palabra de status.

LECTURA DEL REGISTRO DE COMANDO/STATUS



interrupción de timer (este bit se latchea a 1 cuando se alcanza un final de cuenta, se borra leyendo el registro C/S o cuando empieza una nueva cuenta).

Sección de entrada/salida.-

Consiste en cuatro registros descritos a continuación:

- Registro de comando/status (C/S)

Este registro tiene asignado siempre una dirección, que sirve para los dos objetivos: X X X X X 0 0 0.

Cuando se selecciona el registro C/S durante una operación WRITE, se escribe un comando en el registro de comando. El contenido de este registro no es accesible a través de las patas.

Cuando se selecciona el C/S (X X X X X 0 0 0) durante una operación de lectura, la información de status de las puertas I/O y el timer se hacen disponibles en las líneas AD0-AD7.

- Registro PA.

Este registro puede ser programado para ser puertas de entrada o de salida, dependiendo del estado del contenido del registro C/S. También dependiendo del comando, esta puerta puede operar bien en modo básico, o modo permitido (strobed). Las patas I/O asignadas en relación a este registro son PA0-PA7. La dirección de este registro es X X X X X 0 0 1.

- Registro PB.

Funciona igual que el registro PA. Las patas asignadas son PB0-PB7, y la dirección del mismo es X X X X X 0 1 0.

- Registro PC.

Tiene la dirección X X X X X 0 1 1 y contiene sólo seis bits. Los 6 bits pueden programarse para que sean bien ports de entrada, de salida o señales de control para PA y PB programando adecuadamente los bits AD2 y AD3 del registro C/S.

Cuando PC0-PC5 se usa como puerto de control, se asignan tres bits para la puerta A y otros tres para la puerta B. El primer bit es una interrupción que el 8155 envía fuera. El segundo es una señal de salida que indica si el buffer está lleno o vacío, y el tercero es una pata de entrada para aceptar un "strobe" en el modo de entrada permitido.

Para mayor información también recomendamos se acuda a la bibliografía correspondiente. Insistimos, sólo se intenta exponer las características sobre el 8155 que han sido utilizadas en este proyecto, para una mejor comprensión de programas y un mejor mantenimiento.

Sección del timer.-

El timer es un contador de 14 bits que cuenta los impulsos de entrada al timer y proporciona bien una onda cuadrada o bien un impulso cuando se alcanza el final de la cuenta (TC).

El timer tiene la dirección X X X X X 1 0 0 para el byte de menor peso del registro y la dirección X X X X X 1 0 1 para el byte de mayor peso. El direccionamiento del timer sirve para un doble objetivo. Durante una operación WRITE, se carga un registro de longitud de cantidad con una determinada cantidad (en los bits 0-13) y con el modo timer (bits 14-15). Durante una operación READ se leen el contenido del contador (la cantidad actual) y los bits de modo.

Para programar el timer, se carga primero el registro de longitud de contaje, seleccionando las direcciones del timer. Los bits 14-15 seleccionarán el modo de salida del timer. Hay cuatro modos para elegir:

- Sacar un cero durante la segunda mitad de la cuenta.
- Onda cuadrada.
- Unico impulso al alcanzar el TC.

- Unico impulso repetitivo cada vez que alcanza TC y recarga automática del contador al alcanzar TC, hasta que se indique parar mediante la carga de un nuevo comando en C/S.

Los bits 6 y 7 del contenido del registro de comando/status se crean para arrancar y detener el contador. Hay cuatro comandos para elegir:

C/S

BIT

- 0 0 -- NOP - no afecta al funcionamiento del contador
- 0 1 -- STOP - NOP si el timer no ha arrancado. Para la cuenta si el timer no está en marcha.
- 1 0 -- STOP DESPUES DE TC - Para inmediatamente después de que se alcanza el actual TC. (NOP si el timer no ha arrancado).
- 1 1 -- ARRANQUE - Carga el modo y longitud de la cantidad y arranca inmediatamente después de cargarlo (si el timer no está actualmente en marcha). Si el timer está en marcha, inicia el nuevo modo y longitud de la cantidad después de que se alcance el TC actual.

FORMATO DEL TIMER:

Byte más significativo de la longitud de contaje

M2	M1	T13	T12	T11	T10	T9	T8
----	----	-----	-----	-----	-----	----	----

Modo de timer

Byte menos significativo de la longitud de contaje

T7	T6	T5	T4	T3	T2	T1	T0
----	----	----	----	----	----	----	----

M1 y M2 definen el modo del timer como sigue:

<u>M2</u>	<u>M1</u>	
0	0	-- Saca un cero durante la segunda mitad de la cuenta
0	1	-- Onda cuadrada. El periodo de la onda es igual a la longitud de la cantidad programada, con una recarga automática al alcanzar el TC.
1	0	-- Unico pulso al alcanzar el TC.
1	1	-- Recarga automática: un único pulso cada vez que se alcanza TC.

LA USART 8251

Se trata de una interface de comunicación programable: transmisor-receptor sincrónico-asincrónico universal (USART), compatible con los microprocesadores de la familia MCS-85, MCS-48 así como iAPX86,88.

Está integrado en un chip de 28 patillas dual-in-line. Mediante software podemos programar su modo de funcionamiento.

La 8251 acepta datos en forma paralela del bus de datos del sistema y los convierte en una cadena de bits en serie para la transmisión. De igual forma, acepta una cadena de bits serie desde el exterior y los transforma en bytes para pasarlos, mediante el bus de datos, a la CPU.

Su diagrama de bloques y patillaje son los siguientes:

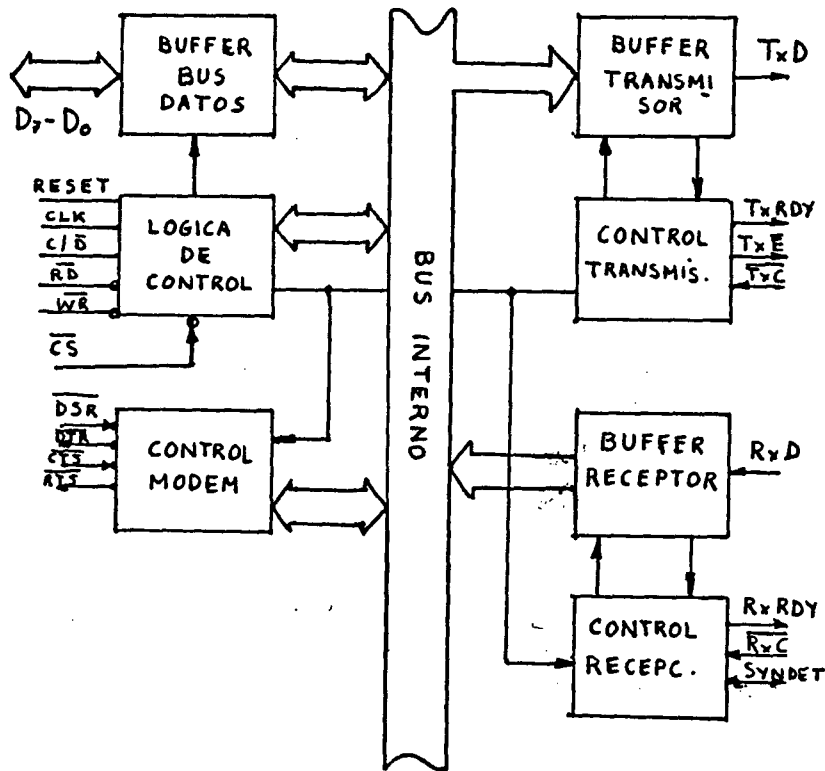
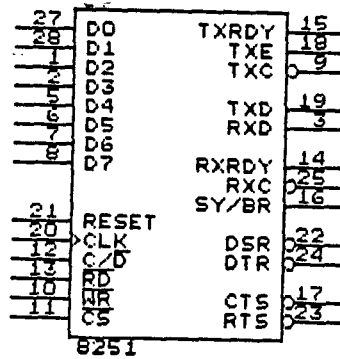


DIAGRAMA DE BLOQUES DEL 8251

A continuación se describe el funcionamiento de este integrado y la forma de programarlo. Se tratará con profundidad la transmisión en modo asincrónico sin modem, por ser el tipo de transmisión/recepción utilizados en este proyecto.

DESCRIPCION FUNCIONAL

Basándonos en el diagrama de bloques, veamos la función de cada uno de ellos:

* Buffer del bus de datos.-

Es un buffer triestado bidireccional de 8 bits. Los datos son transmitidos o recibidos a través de él, mediante las instrucciones IN y OUT de la CPU, además de los bytes de control y comando para programar el modo y características de la comunicación.

* Reset.-

Un uno a través de esta patilla de entrada permite el reseteo de la USART, y permanecerá inactiva hasta que sea nuevamente programada.

* Clock.-

La entrada CLK se usa para generar un secuenciamiento interno de dispositivo. No hay referenciadas para este CLK ninguna entrada o salida externas, pero la frecuencia del mismo debe ser al menos 30 veces mayor que las velocidades

de recepción y transmisión.

* Write (WR)..-

Un nivel bajo en esta señal le informa de que la CPU está escribiendo en ella una palabra de control o un dato.

* Read (RD)..-

De igual forma que el anterior, le indica a la USART que la CPU va a leer un dato o su estado.

* Control/dato (C/D)..-

Junto con READ y WRITE, informa a la 8251 si se está realizando una lectura/escritura de un dato (nivel bajo) o de una palabra de comando (nivel alto).

* Chip Select (CS)..-

Selecciona la USART para su direccionamiento y transferencia de datos.

* Control de modem..-

Cuenta con una serie de entradas y salidas de control

usadas para la interface con un modem. Así tiene:

- DSR (data set ready): datos preparados.
- DTR (data terminal ready): terminal de datos preparado.
- RTS (request to send): petición de envío.
- CTS (clear to send): listo para enviar.

NOTA: Este protocolo de comunicación no ha sido utilizado en nuestro prototipo, por no existir la necesidad de una comunicación a través de modem. Es por eso que simplemente se nombran, y se sugiere acudir a las características del integrado si se desea más información sobre dichas patillas.

* Buffer transmisor.-

Este buffer acepta datos en formato paralelo desde el buffer del bus de datos, los convierte en formato serie, inserta los bits o caracteres auxiliares necesarios para la transmisión y los transmite a través de la salida TxD.

* Control de transmisión.-

Es el bloque encargado de manejar las actividades asociadas con la transmisión de datos en serie, proporcionando unas señales que nos ayudan a conocer el

estado del transmisor:

- TxRDY: Indica a la CPU que el transmisor está preparado para aceptar datos desde ella. El pin TxRDY puede usarse para interrumpir el microprocesador con el objeto de indicarle esta situación, o bien la CPU podrá saber este estado por software.
- TxE: A nivel alto indica que la 8251 no tiene caracteres que enviar, y se resetea cuando recibe un carácter de la CPU.
- TxC: Reloj del transmisor: es la entrada del reloj que controla la transmisión. La palabra de modo de transmisión permite realizar una división de la señal de reloj en tres formas: 1/1, 1/16, y 1/64.

* Buffer receptor.-

Acepta datos en serie desde el exterior y los convierte en formato paralelo, separando los bits utilizados como técnica de comunicación. Los datos son recibidos por la entrada RxD.

* Control de recepción.

Lleva todas las tareas relacionadas con la recepción:

- RxRDY: Salida que indica a la CPU que ha sido recibido un carácter y que está preparado para ser introducido en ella. La CPU puede detectar este estado mediante hardware (conectando RxRDY a una entrada de interrupción) o mediante software, testeando el byte de status de la 8251. En el modo asíncrono, el receptor debe detectar los bits de start y stop, aparte del byte transmitido, para que la salida RxRDY pueda mandar un aviso de dato recibido.

- RxC: Clock del receptor: es la entrada para el reloj que controla la recepción. Al igual que TxC, esta frecuencia puede ser dividida por un factor: 1, 1/16 o 1/64. Normalmente las entradas TxC y RxC se puentean y se les aplica la misma señal de reloj.

PROGRAMACION DE LA 8251

Mediante las palabras de modo y comando enviadas por la CPU a la USART se puede programar distintos aspectos de su forma de comunicación: velocidad de transmisión, longitud del carácter, número de bits de stop, operación asíncrona o síncrona, paridad par o impar, o ninguna,...

Después de la programación, la USART queda preparada para la comunicación. Existen dos formatos de programación:

- Instrucción del modo.
- Instrucción de comandos.

La instrucción de modo define las características operacionales de la 8251. Esta instrucción siempre va seguida de una operación de reset (interno o externo).

La instrucción de comando viene después de la anterior, y controla la operación de la 8251. La figura siguiente indica una secuencia típica de datos en modo asíncrono:

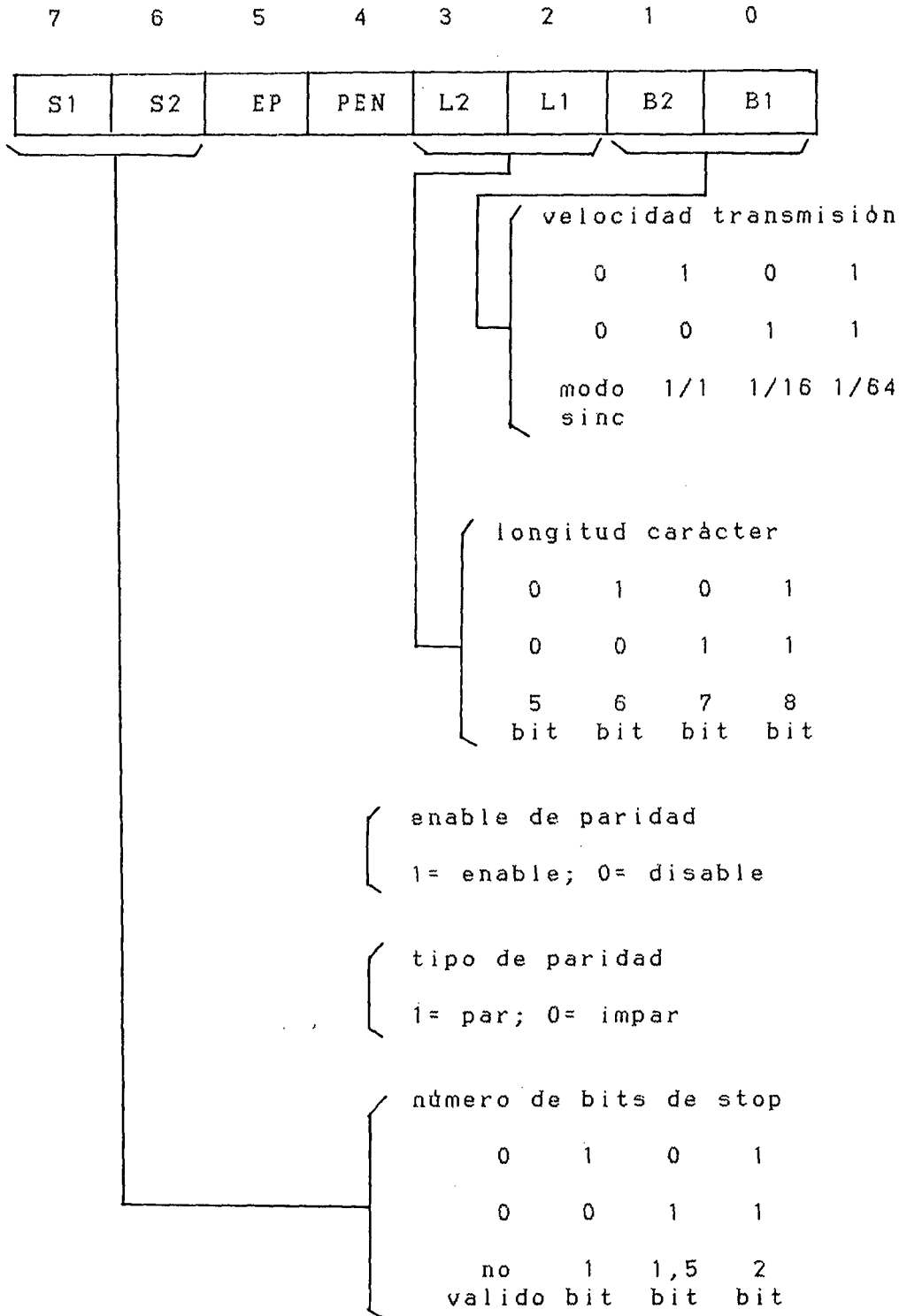
C/D= 1	RESET
C/D= 1	INSTRUCCION DE MODO
C/D= 1	INSTRUCCION DE COMANDO
C/D= 0	DATOS
C/D= 1	INSTRUCCION DE COMANDOS
C/D= 0	DATOS
C/D= 1	INSTRUCCION DE COMANDOS

La instrucción de comandos puede ser escrita en cualquier momento durante la operación de la USART, mientras se reciben o transmiten datos.

* Instrucción de modo.-

Esta palabra que define el modo de funcionamiento sólo puede ser enviada desde la CPU a la USART después de resetear ésta. Hay dos formatos distintos para este byte de modo, según se trate de una transmisión síncrona o asíncrona. El esquema siguiente corresponde al formato de

la instrucción de modo en comunicación asíncrona:



* MODO ASINCRONO.-

1) Transmisión en modo asíncrono.

Cuando la CPU envía un carácter a la USART, ésta automáticamente añade un bit de start (nivel bajo), seguido de los bits de datos (primero el bit menos significativo), y después el número de bits de stop programados.

También se inserta antes del bit de stop el bit de paridad par o impar, en caso de programarse con paridad, según se defina en la instrucción de modo. Después de esto se transmite el dato en serie por la salida TxD.

SALIDA DEL TRANSMISOR

TxD	BIT START	BITS DATOS	BIT PARIDAD	BITS STOP
-----	-----------	------------	-------------	-----------

FORMATO DE TRANSMISION

BYTE DE CPU (5 A 8 BITS POR CARACTER)

CARACTERES DE DATOS

SALIDA SERIE DE DATOS

BIT START	DATOS	BIT PARIDAD	BITS STOP
-----------	-------	-------------	-----------

2) Recepción en modo asincrono.

En reposo, la entrada RxD está a nivel alto. Cuando se detecta un flanco de bajada es señal de que ha llegado un bit de start. Para tomar como válido este bit se chequea otra vez el nivel bajo en el centro del bit (en los modos 1/16 y 1/64 solamente); si persiste el nivel bajo la USART lo tomará como un bit de start válido, y no como un ruido. A partir de este momento un contador de bits empezará a contar los bits que se van recibiendo; localiza los bits de datos, de paridad y de stop, y los chequea en su centro.

Si aparece un error de paridad se pondrá a uno el flag de error de paridad. El bit de stop indica el final del carácter, siempre a nivel alto. Si se detecta un nivel

bajo como bit de stop aparecerá un error de framing en el flag correspondiente.

El carácter recibido se carga en el buffer del bus de datos, la patilla RxDY se pone a uno para indicar a la CPU que ya puede ser leído. Si habla antes en el buffer otro carácter que no fue leído aparecerá un error de overrun. Todos los flags de error se pueden resetear con una instrucción de reset de errores. La aparición de estos errores no afectará a la operación de la USART.

ENTRADA DEL RECEPTOR

RxD	BIT START	BITS DATOS	BIT PARIDAD	BITS STOP
-----	-----------	------------	-------------	-----------

longitud
dependiente de programación

no aparece en el bus
de datos

FORMATO DE RECEPCION

ENTRADA SERIE DE DATOS

BIT START	DATOS	BIT PARIDAD	BITS STOP
-----------	-------	-------------	-----------

BYTE CPU (5-8 BITS POR CARACTER)

CARACTERES DE DATOS

cada momento. Mientras que las instrucciones de modo y comando son introducidas en la USART con una instrucción IN de la CPU, y con C/D a nivel lógico 1, el estado puede ser conocido con una lectura: C/D a nivel uno y la instrucción OUT de la CPU. Con ello se transfiere un byte de información al microprocesador, byte que tiene la siguiente interpretación:

FORMATO DE LECTURA DEL STATUS

7	6	5	4	3	2	1	0
DSR	SYNDET BRKDET	FE	DE	PE	TxEMPTY	RxRDY	TxRDY

TxRDY: se pone a uno si el buffer de datos está vacío, CTS= 0 y TxEN= 1.

RxRDY, TxEMPTY, SYNDET-BRKDET: la misma definición que sus pines.

PE (error de paridad): se pone a uno cuando es detectado un error de paridad.

OE (error de overrun): se pone a uno cuando la CPU no ha sido leído un carácter y el buffer se llena con otro nuevo.

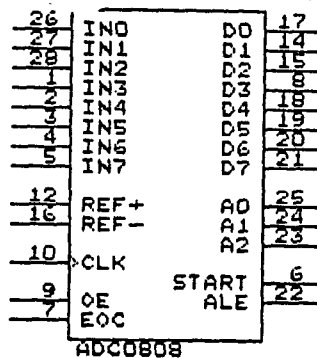
FE (error de framing): en transmisión asincrónica, este

flag se pone a uno cuando no se ha detectado un bit de stop.

En las características de la USART 8251 se puede realizar un estudio más profundo de este integrado. Sin embargo, con lo expuesto aquí se dispone de la suficiente información para entender cómo hemos utilizado en este proyecto la USART.

EL CONVERTIDOR ADC0808

El ADC0808 es un convertidor analógico-digital compatible con microprocesadores de 8 bits, con ocho canales de entrada analógica multiplexados.



FUNCIONAMIENTO

El dispositivo contiene ocho canales de entrada analógica multiplexados. Cada entrada es seleccionada mediante un decodificador de direcciones. Posee tres entradas de direccionamiento: A, B y C, siendo la patilla

A la de menor peso. La selección de cada entrada analógica se realiza según la tabla siguiente:

ENTRADA ANALOGICA	DIRECCION		
	C	B	A
IN0	0	0	0
IN1	0	0	1
IN2	0	1	0
IN3	0	1	1
IN4	1	0	0
IN5	1	0	1
IN6	1	1	0
IN7	1	1	1

El latcheo de la dirección se aplica un uno a la señal ALE.

CARACTERISTICAS DE LA CONVERSION

El convertidor está dividido en tres secciones principales:

-, La red en escalera 256R.

- el registro de aproximaciones sucesivas.
- el comparador.

La red en escalera 256R se trata de un conjunto de 256 resistores colocados en serie, todos ellos con la misma resistencia, y además un array de conmutadores para la elección del voltaje apropiado. Esta estructura supone una ventaja con respecto a otras como las escaleras R/2R, pues al ser todos los resistores idénticos e integrados en una misma pastilla, las variaciones en el valor de las mismas no afectará al conjunto, y la conversión seguirá siendo fiable en un amplio margen de temperatura.

El registro de aproximaciones sucesivas (SAR) realiza ocho comparaciones hasta determinar el voltaje de entrada, tantas como bits posee el dato digital de salida.

La conversión comienza en el flanco de bajada del pulso de comienzo de conversión (señal START). La salida de la señal EOC (final de conversión o end of conversion) pasará a nivel bajo después de recibirse el flanco de subida de la señal de START (como máximo ocho ciclos de reloj después)

Las entradas $V_{ref(+)}$ y $V_{ref(-)}$ soportan un máximo de

diferencia de potencial de 5 V. Normalmente se colocará $V_{ref(+)}$ a cinco voltios y $V_{ref(-)}$ a masa; con estos valores se tendrá una precisión de 20mV, es decir, el bit menos significativo (LSB) tiene un valor de 20mv (o lo que es lo mismo, una entrada analógica de 20mV corresponderá a un dato digital de 01H).

También es posible tomar una tensión de referencia menor de cinco voltios: por ejemplo, $V_{ref(+)} = 3,75 \text{ V}$ y $V_{ref(-)} = 1,25 \text{ V}$, lo que daría una diferencia de potencial de 2,5 V. En este caso el valor del LSB será la mitad que para una diferencia de 5 V. Lo que se debe tener siempre en cuenta es que la diferencia entre $V_{ref(+)}$ y $V_{ref(-)}$ no puede superar los 5 V.

Además, es necesario que esta tensión sea lo más estable posible, pues un voltaje que oscile nos proporcionaría unos datos digitales erróneos. La utilización de un regulador de tensión es bastante aconsejable.

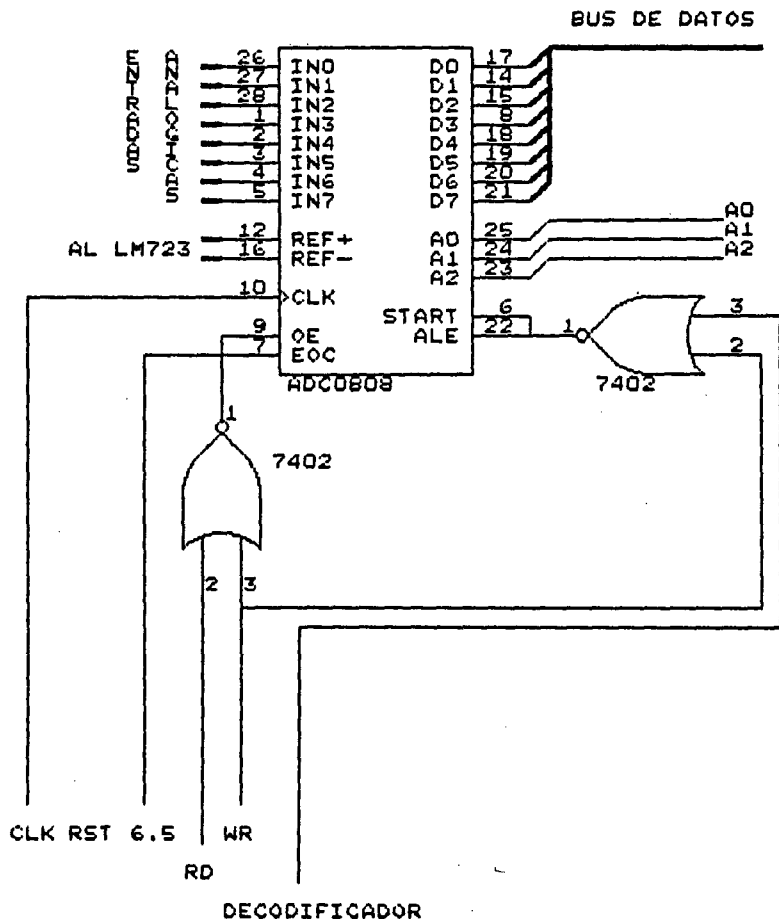
La patilla CLK es la entrada de una señal de reloj que utilizará el ADC para realizar su trabajo. Este reloj puede tener una frecuencia cualquiera dentro del margen establecido: 10KHz a 1280KHz, con un valor típico de 640KHz. Para este valor de frecuencia, el tiempo de

conversión es de 100 microsegundos, pudiendo llegar a un máximo de 116 microsegundos, en el caso peor.

La salida EOC (final de conversión) toma el nivel cero mientras se lleva a cabo la conversión. Al término de ésta cambia al estado 1 para indicar dicha finalización. El dato digital es depositado en un latch de salida; para leerlo habrá que habilitar el latch colocando un uno en la entrada OE (output enable).

La conexión del ADC0808 a un sistema microprocesador puede realizarse de dos formas:

- 1) Introduciendo los datos por un puerto. Se conectan las señales de control del convertidor (ALE, START, OE) y las las patillas de direccionamiento (A, B y C) a un puerto de salida del sistema, y las ocho salidas digitales (DB0-DB7) a un puerto de entrada. Con instrucciones de input y output se maneja el convertidor.
- 2) Conectando el 0808 directamente al bus del sistema, como un módulo más enganchado al bus. El acceso al convertidor sería idéntico a un acceso a memoria. Un esquema práctico para ello es el de la figura:



También existen dos formas de averiguar cuándo acaba una conversión para inmediatamente comenzar la siguiente:

- 1) Dependiendo del tiempo de conversión: para el ejemplo de una frecuencia de trabajo de 640KHz el tiempo de conversión es de 116 microsegundos como máximo. Basándonos en esto, una conversión la iniciáramos tras ejecutar una rutina de retardo de

116 microsegundos en el microprocesador. Si utilizamos una frecuencia distinta a la indicada, el tiempo de conversión puede ser calculado perfectamente en un osciloscopio, midiendo el tiempo que la señal de EOC permanece a nivel bajo en una conversión.

- 2) Por interrupción: conectando la patilla EOC a una entrada de interrupción del microprocesador podremos iniciar una segunda conversión. La subrutina de servicio de interrupción correspondiente se encargará de ello.

Los métodos más eficaces, en cuanto al ahorro de tiempo y mejor aprovechamiento de las capacidades del microprocesador es el de interrupción y conexión al bus.

I N F L U E N C I A D E L O S

C A M P O S

E L E C T R O M A G N E T I C O S

La puesta en funcionamiento de un sistema basado en microprocesador como control de estabilidad de una central eléctrica acarrea serios problemas. Los alternadores sincrónicos que alimentan a las líneas eléctricas son de gran potencia, y generan fuertes campos electromagnéticos a su alrededor. Esto supone para nuestro sistema microprocesador tener que estar sometido a dichos campos, con fatales consecuencias para él. La fuerza de los campos inducidos es tal que pueden incluso borrar la EPROM del sistema microprocesador, convirtiéndolo en un montón de silicio inútil.

Es por esta razón que se hace imprescindible situar el sistema microprocesador lo más alejado de los

alternadores, con el fin de evitar la influencia de los campos electromagnéticos; la comunicación entre sistema de control y generadores debe ser a distancia. Pero es más, aunque se establezca esa distancia los hilos de conexión entre los alternadores y el sistema son también fuente de interferencias e inducciones. Los ruidos que llegan al sistema a través de la masa son bastante dañinos e inutilizan el sistema de control, así que es imprescindible un desacoplamiento total entre las masas de los alternadores y las del sistema.

En realidad, las medidas sobre la potencia que en cada momento está dando cada generador, así como la potencia consumida por cada línea, llegan al sistema a través de unos convertidores analógicos-analógicos. Estos convertidores suministran al sistema una corriente entre 4 mA y 20 mA, que corresponden a unas potencias de 0 (generador o línea desconectado) y de 9400 KVA, respectivamente. Por tanto, el aislamiento de masas a que nos referimos es al de los convertidores analógicos-analógicos y nuestro sistema.

El aislamiento de tierras no es suficiente para solucionar el problema de las inducciones y ruidos. El campo electromagnético sigue influenciando en el sistema, a menos que se consiga un apantallamiento total del mismo.

La protección contra estos campos es importante, como dijimos, pues su acción es capaz de destruir el sistema de control.

A continuación se dan algunas definiciones y formas de llevar a cabo la inmunización contra los campos en sistemas eléctricos. A este fenómeno lo llamamos apantallamiento.

APANTALLAMIENTO

Como se dijo, un posible problema que se nos plantea al montar un equipo en una central eléctrica es el debido a los campos eléctricos y magnéticos que se crean alrededor.

La radiación electromagnética está compuesta de dos partes inseparables: un campo eléctrico E y un campo magnético H . Estos campos complementarios varían en intensidad y polaridad según la frecuencia.

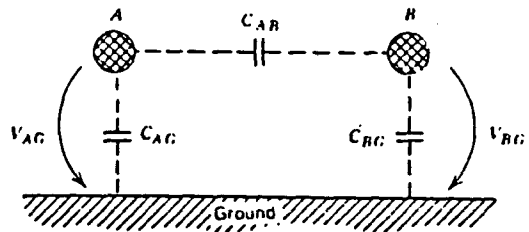
Afortunadamente, cuando consideramos interferencias electromagnéticas, la intensidad disminuye a medida que nos alejamos de la fuente de interferencias. La característica del campo eléctrico se mide en voltios por metro, y el campo magnético en amperios por metro.

Campos electrostáticos.-

Los campos electrostáticos son producidos por oscilaciones en las cargas. El mecanismo de acoplamiento electrostático es a través de condensadores conjuntos. La ausencia de apantallamiento da lugar a inducciones. Por ejemplo, supongamos dos torretas, A y B conectadas a masa. Si se aplica una tensión entre A y masa se inducirá otra entre B y masa, a causa de una capacidad creada entre estos dos puntos. El valor de dicha tensión es:

$$V_{BG} = V_{AG} \left(\frac{C_{AB}}{C_{AB} + C_{BG}} \right)$$

donde V_{BG} es la tensión entre B y masa, V_{AG} la tensión entre A y masa, C_{AB} la capacidad entre A y B, C_{BG} la capacidad entre B y masa.

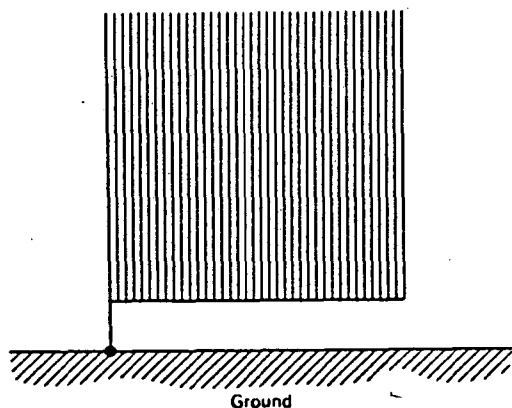


Apantallamiento electrostático.-

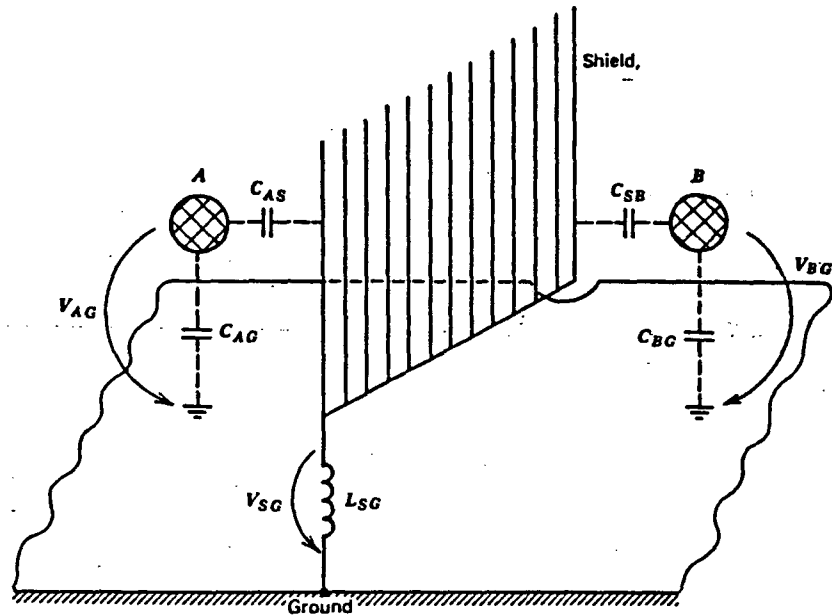
Una malla de alambre en buen contacto con la tierra

puede ser suficiente para proporcionar un buen apantallamiento electrostático; el efecto pantalla de este apantallamiento metálico no es afectado por los agujeros y estrechas ranuras.

Se usa una malla de alambres paralelos verticales si se necesita apantallamiento contra campos electrostáticos de baja frecuencia.



Sin embargo, para altas frecuencias la impedancia de la pantalla, o lo que conecta la pantalla a la tierra, podría ser comparable con la impedancia de la capacidad entre la pantalla y la fuente de radiación. En tal caso el potencial de la pantalla no es mayor que cero, y no suministra el aislamiento requerido, teniendo que añadir al esquema de baja frecuencia una bobina para su aplicación en altas frecuencias.



Cuanto menor sea la distancia entre el chasis y la tapa (cover), menor apantallamiento se necesitará, ya que cualquier posible campo generado se absorberá por ellos.

Campos magnéticos

Los campos magnéticos generalmente son producidos por movimientos de cargas de partículas. Estos campos están alrededor de los conductores que llevan corriente. La corriente que circula por el circuito produce un flujo que induce tensión en otro circuito.

El apantallamiento de equipos contra campos magnéticos es costoso y problemático a bajas frecuencias (entre 10 Hz y 3 KHz); en este caso, la atenuación del campo depende principalmente de la pérdida de absorción del material apantallado. En altas frecuencias el material de apantallamiento proporcionará una disminución del campo magnético, que es lo más adecuado en la mayoría de los casos prácticos. Los problemas que pueden surgir vienen de pequeñas perturbaciones introducidas por medio de discontinuidades como agujeros, ranuras,...

Apantallamiento electromagnético.-

Se está trabajando en soluciones para el apantallamiento. Este depende de:

- La naturaleza del campo magnético o eléctrico.
- La frecuencia del campo.
- La distancia al punto de radiación de ruido.
- El aislamiento dado por el espacio entre el punto de radiación y el punto del circuito sensible a la radiación (en ausencia de campo o con campo conocido a considerar).

La efectividad de un apantallamiento es la característica que indica cuán preparado está el conductor para prevenir la propagación de energía electromagnética por el canal. La efectividad del apantallamiento es función de:

- la frecuencia e impedancia presentada al campo incidente.
- la distancia desde la fuente de radiación.
- parámetros tales como el grosor, permeabilidad y conductividad del conductor protector.

El hecho de que haya discontinuidades en el apantallamiento dará lugar a la fuga de ruidos a través de las discontinuidades, que es un grave problema para los diseñadores de equipos de radiofrecuencia.

El metal utilizado en la fabricación de chasis es el aluminio, ya que es barato, ligero y también tiene excelentes características de apantallamiento en campos electrostáticos.

La falta de apantallamiento generará ruido, cuyas principales fuentes son:

- Defecto en el diseño de circuitos, tal como fugas de

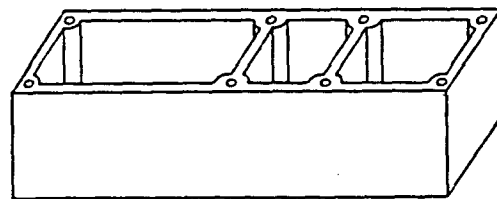
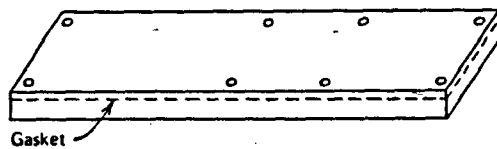
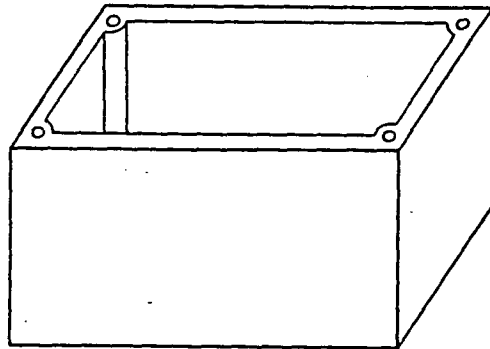
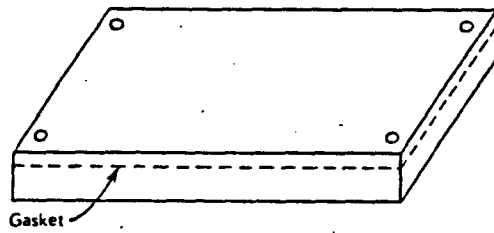
acoplamiento que generarán acoplamientos a través de la línea de alimentación.

- Técnicas de montaje de circuito impropias, como una tierra pobre.
- Acoplamiento inadecuado.

Los objetivos a tener en cuenta a la hora de conseguir un buen apantallamiento se relacionan a continuación:

- Eliminar cualquier tipo de señal extraña, incluso aquellas cuya frecuencia cae por fuera de la banda del circuito, ya que ésta se podría trasladar de banda mediante alguna conversión, amplificándose, y dando lugar a mezclas y posibles oscilaciones.
- En esquemas que trabajen con diferentes frecuencias y una misma fuente de alimentación, lo mejor sería dividirlo en circuitos de diferentes anchos de banda y filtrar cada circuito, siendo siempre preferible filtros tipo RC a los LC.
- En un esquema con parte analógica y parte digital es conveniente usar fuentes separadas para prevenir el alargamiento de transitorios en los circuitos digitales por causa de los circuitos analógicos.

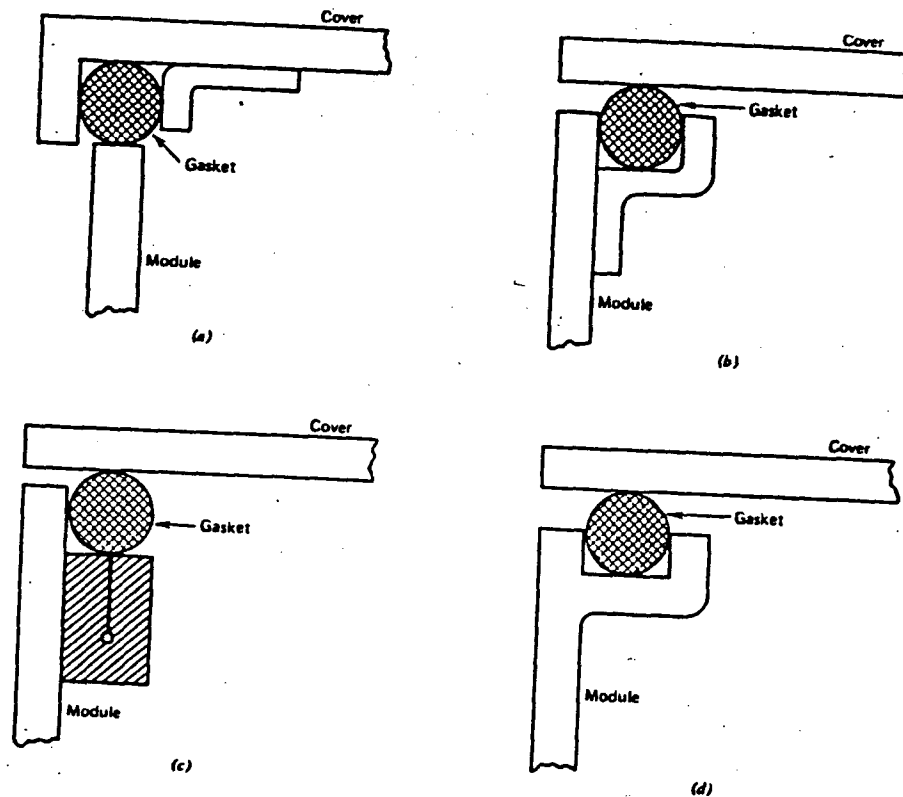
- Situar la alimentación del transformador lo más alejada posible, para minimizar picos de frecuencia en la línea.
- Apantallamiento de circuitos internos para prevenir posibles fugas.
- Aislamiento de diferentes circuitos con el fin de evitar realimentaciones y acoplamientos indeseados.
- Filtrar la línea de potencia para atenuar la propagación de señales de radiofrecuencia.
- Conseguir baja impedancia en tierras.
- Realizar un diseño con distribución modular, es decir, por compartimentos. Este método se utiliza comúnmente contra campos electrostáticos, magnéticos y electromagnéticos. La distribución puede consistir en un solo módulo, cuando se requiere un aislamiento del ambiente exterior, o de múltiples compartimentos, cuando se desea un aislamiento entre distintas partes del mismo sistema.

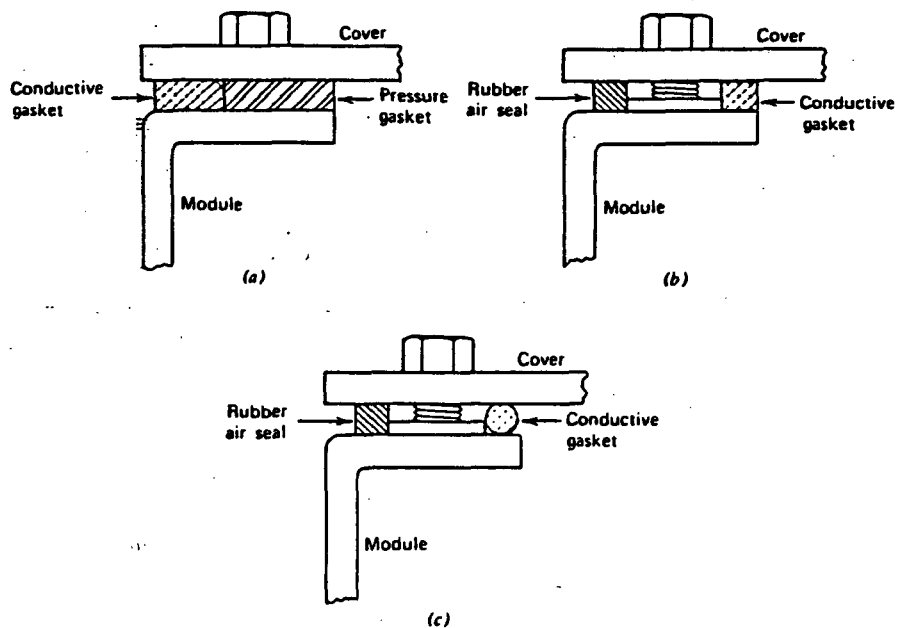


Para la modulación existen cuatro tipos de procesos: milling, casting, brazing y soldering, dependiendo del grado de aislamiento requerido. El primero da un muy alto

grado de aislamiento, pero el coste también es bastante elevado; se usa para trabajar en prototipos. Para alta calidad se usa el de casting. El proceso brazing se asocia con el aluminio. El soldering es el más barato, pero el de peor calidad, presentando problemas referentes al grosor máximo del material a soldar.

- Por último, el número de tornillos de la caja deberá ser mínimo, además de recomendarse un encajador para evitar la entrada de ruidos a la caja.





ULTIMOS DETALLES

En el caso práctico que nos ocupa, un sistema microprocesador de control de estabilidad de una central eléctrica, cabe una posibilidad a la hora de tratar de evitar la influencia de los campos electromagnéticos.

La cercanía del sistema a la central es bastante problemática. Fácilmente se ve afectado por las perturbaciones mencionadas y es necesaria una protección. Además de todo lo comentado para realizar el

apantallamiento adecuado (separación galvánica de masas,...) sería conveniente situar el sistema lo más alejado posible de la central, llevando a cabo un telecontrol de la misma.

Para ello UNELCO cuenta con diversos medios. Hoy en día esta empresa tiene instalado en la mayoría de las subestaciones de la red eléctrica equipos de enlace a través de la propia línea de transporte. Posee una configuración en baja frecuencia, con varias posibilidades:

- Posibilidad de telefonía privada selectiva, o punto a punto.
- Posibilidad de transmisión a través de un canal de 600 baudios, o bien de 1200 baudios, en forma sincrónica, destinado al telecontrol.
- Posibilidad de transmisión a través de otro canal de 110 baudios que bien podría destinarse al control de la estabilidad.

La comunicación remota "sistema de control-central" es una buena forma de evitar muchas perturbaciones.

P A R T E I I I

M A N U A L D E M A N T E N I M I E N T O
S I M U L A C I O N E N T I E M P O R E A L

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

A la hora de entregar al usuario un producto terminado es necesario adjuntarle un manual de mantenimiento del mismo. Esta necesidad será tanto más grande cuanto mayor complejidad de funcionamiento y manejo posea el producto.

Esta última parte del proyecto pretende ser un manual de documentación y mantenimiento del sistema de control para la central eléctrica.

Expondremos varios puntos, tal como son los siguientes:

* Teoría de funcionamiento.-

Para facilitar el conocimiento del sistema a un posible reparador del mismo resulta interesante ofrecer en pocas palabras su filosofía de funcionamiento, tanto software como hardware. Unos diagramas de bloques generales, así como, organigramas, ilustran de forma rápida lo que es el

sistema y la función que realiza. Para un técnico experimentado estos esquemas bastan. Sólo con su estudio podría saber qué función tiene cada componente del sistema.

Por este motivo se presentan aquí los diagramas de bloques software y hardware generales, aunque ya hayan sido expuestos de forma más detallada en secciones anteriores.

* Características y limitaciones del sistema.-

Puede suceder que el usuario de un sistema crea haber encontrado una avería cuando dicho sistema no realiza las funciones que se espera. Y es posible también que tal hecho no constituya una verdadera avería. Quizás el usuario no sepa cuál es la capacidad del dispositivo, ni las limitaciones que posee.

Como consecuencia, se considera importante exponer estos aspectos y dejar bien claro para qué está diseñado el sistema de control.

* Pruebas de mantenimiento.-

La ejecución de varias pruebas de comprobación del buen

funcionamiento del sistema son imprescindibles en su puesta a punto y mantenimiento constante.

Según los resultados obtenidos al someter al sistema a un cierto examen, el reparador o mantenedor podrá saber si algo falla y dónde falla, de manera rápida y efectiva.

* Posibles ampliaciones y mejoras.-

A medida que se ha diseñado e implementado el sistema han surgido nuevas ideas para mejorar su capacidad de control, tal como se comentará.

* Simulación en tiempo real.-

Por último se explicará la forma en que se ha llevado a cabo la comprobación y puesta a punto del prototipo realizado. La simulación permite a los diseñadores comprobar su producto, presentarlo al cliente en una primera fase, y ayudar a la detección de errores en cualquier momento de la vida activa del dispositivo.

T E O R I A D E F U N C I O N A M I E N T O

En este apartado se exponen, como se indicó, los diagramas de bloques del funcionamiento hardware y del funcionamiento software.

ORGANIGRAMA SOFTWARE

Tras la inicialización del sistema, el programa testea constantemente la estabilidad del sistema. En realidad, la toma de datos por parte del sistema se realiza de forma periódica, a través de interrupción hardware, y se produce con esta frecuencia fija, independientemente de las otras tareas que se ejecuten.

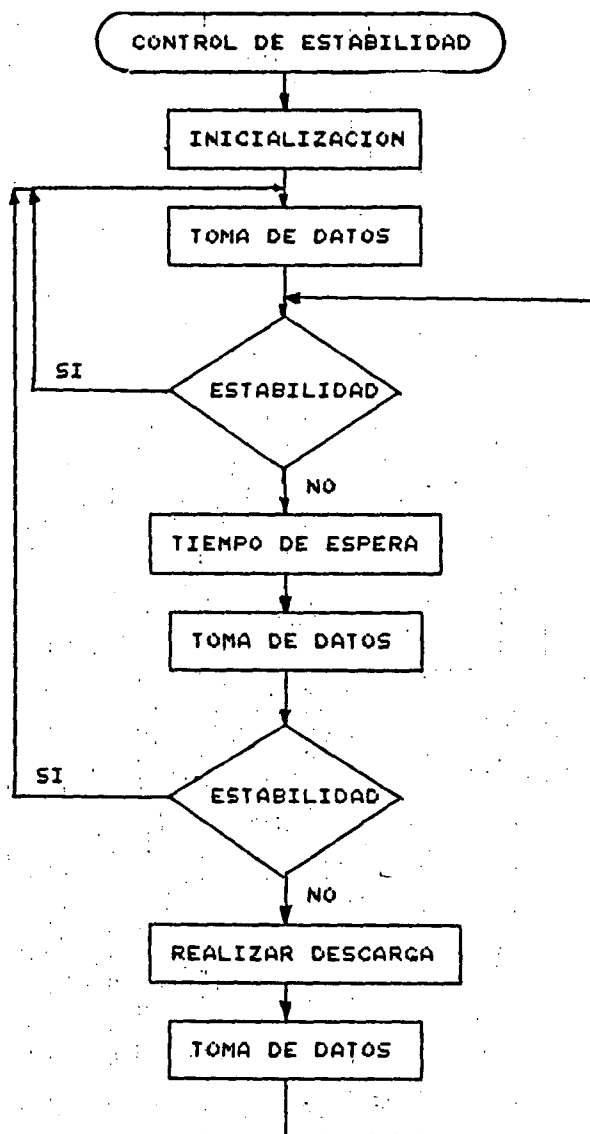
Además se realizan algunas transmisiones de datos hacia el ordenador después de la ocurrencia de alguna descarga. Se produce también varias veces durante la ejecución del tiempo de espera que se asigna después de detectar una situación de posible inestabilidad y antes de decidir la descarga. Otro momento en que se producen ejection

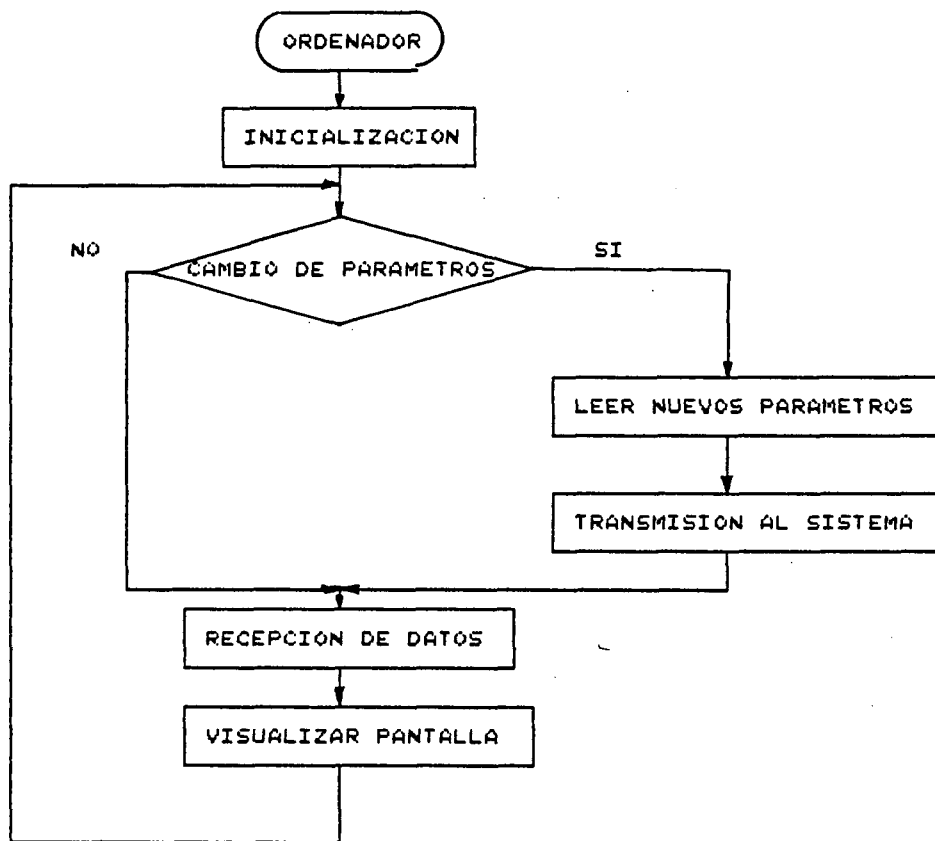
transmisiones es durante el bucle normal de ejecución, destinado a la vigilancia continua del sistema.

El programa del PC se encarga de recoger los datos sobre las corrientes desde el sistema microprocesador y visualizarlos, además de activar alarmas visuales y sonoras en caso de que alguna de las potencias sea crítica, o algún elemento se encuentre desconectado. Además cuenta con las posibilidades de variar los parámetros del sistema, en caso de que el usuario lo desee, o congelar los datos que aparecen en pantalla para realizar un estudio más cómodo de los mismos.

Otra misión del ordenador es permitir introducir por teclado nuevos parámetros, a solicitud del usuario, y enviarlos al sistema para que los procesamientos se hagan con referencia a esos nuevos parámetros.

En las páginas siguientes se presenta el organigrama software general del sistema microprocesador y del PC, además de una figura ilustrativa de la forma en que el ordenador presenta los datos en pantalla.





		CONTROL DE ESTABILIDAD		EG= 25440 e IL= 25439	
	POTENCIA		POT.NOMINAL		ESTADO
Generador 1:	8724 KVA	9.400	KVA	ON	
Generador 2:	8105 KVA	9.400	KVA	ON	
Generador 3:	8611 KVA	9.400	KVA	ON	
	POTENCIA				ESTADO
Línea 1:	5333 KVA			ON	
Línea 2:	6436 KVA			ON	
Línea 3:	4843 KVA			ON	
Línea 4:	3678 KVA			ON	
Línea 5:	5149 KVA			ON	

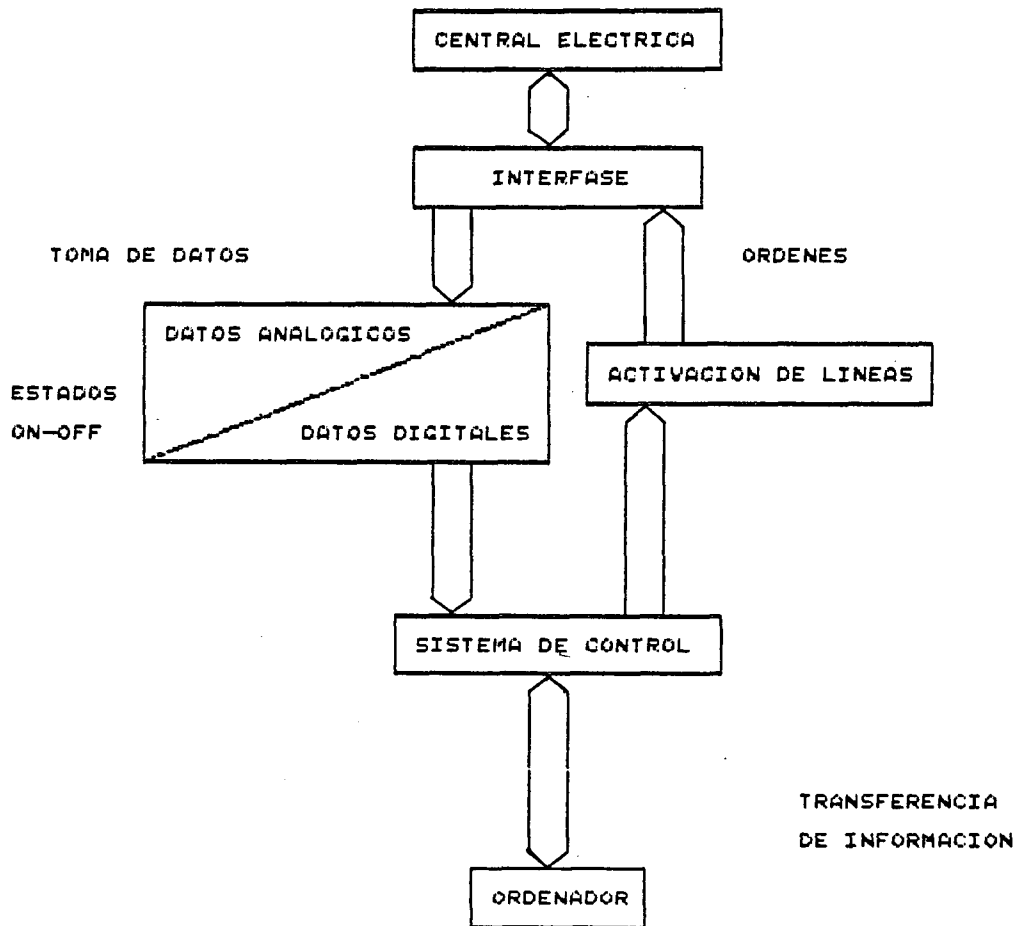
DIAGRAMA DE BLOQUES HARDWARE

El sistema microprocesador de control mantiene un doble contacto, según se aprecia en el diagrama de la página siguiente:

- Por una parte con la central eléctrica, recogiendo datos sobre la misma, y enviando órdenes de actuación. Es lo que forma el módulo de "toma de datos-envío de órdenes".
- Por otra parte mantiene contacto con el ordenador, transmitiendo los datos recogidos y aceptando otros de él, como son valores de potencias nominales en los generadores que el usuario desea introducir. Es el módulo de comunicación con el PC.

A todo esto se le suma la labor de procesamiento de datos que debe realizar de forma continua, a partir de los valores recibidos, y la toma de decisiones. Esta función la efectúa el módulo de CPU.

DIAGRAMA GENERAL HARDWARE



En realidad, la teoría de funcionamiento ha sido expuesta minuciosamente en otras secciones; en este apartado se pretende simplemente dejar constancia de la necesidad de dar al usuario o reparador una idea rápida, general e intuitiva de la misión del sistema.

C A R A C T E R I S T I C A S
Y
L I M I T A C I O N E S D E L S I S T E M A

Las características del sistema han quedado reflejadas en anteriores exposiciones, tales como las descripciones hardware y software. Pero es interesante comentar algunos aspectos limitadores.

Uno de ellos quizás sea lo referente al tiempo. La relativa lentitud del microprocesador 8085, en comparación con otros microprocesadores disponibles en el mercado, sumado a sus características de funcionamiento, con datos de sólo ocho bits, supone una barrera en cuanto a la precisión que se podría alcanzar en el control de la central.

Estas mismas características limitadoras las posee el convertidor A/D utilizado. No obstante, la velocidad de

ejecución para nuestro caso particular es suficientemente buena. Proporciona un control adecuado; aunque podría mejorarse con el empleo de otros circuitos más sofisticados, la precisión que ofrecen es aceptable.

El software y hardware desarrollados se adaptan a un esquema de central eléctrica concreto: una central formada por tres grupos trabajando en configuración paralela, y suministrando carga a cinco líneas. La ampliación de estas características requeriría una serie de modificaciones en el hardware y software, aunque la filosofía de funcionamiento no sufriría variaciones, y los cambios necesarios tampoco supondrían grandes quebraderos de cabeza.

Existe una serie de eventos que el sistema no es capaz de detectar, aunque lo podría hacer con unas simples alteraciones y ampliaciones hard-soft, o muchas veces sólo de tipo soft. Se trata, por ejemplo, de la detección de posibles errores o problemas deducibles de los datos tomados, como cortocircuito en alguna línea; o fallos en la desconexión de líneas, descarga consecutiva de dos o más líneas (circunstancia poco frecuente en la realidad), la conversión de un generador en motor, toma de datos errónea,...

En principio, tampoco hay implementado un esquema para conmutación de placas en caso de averías. No se dispone de los medios para comprobar el buen funcionamiento de ciertos circuitos como el convertidor, o artilugios que permitan al mantenedor comprobar que las tomas de datos son fiables.

En cuanto a la comunicación del sistema de control con el ordenador, la distancia entre éstos no debe ser mayor de 15 m., ya que no está diseñado para la utilización de un modem; simplemente está pensado para comunicaciones a través de un sencillo cable coaxial.

Otra limitación del sistema es su enorme exposición a fuertes campos electromagnéticos que pueden inutilizarlo, por lo que se hace preciso la utilización de un apantallamiento adecuado en el momento de su instalación en el lugar de trabajo definitivo.

El establecimiento de las comunicaciones a una velocidad de 1200 baudios es otro aspecto limitador. Esta velocidad es relativamente pequeña y el sistema consume demasiado tiempo en las transmisiones, aproximadamente 80 ms. La velocidad podría aumentarse a 2400 o 9600 baudios, pero la probabilidad de errores en la comunicación también se incrementaría. Es por ello que se

opta por la lentitud a cambio de la fiabilidad.

El sistema está preparado para informar cada cierto tiempo al ordenador de los datos de los datos recogidos desde la central. No todas las muestras tomadas por el sistema son transmitidas al ordenador; es mucho mayor el número de veces que los datos son actualizados en el sistema que el número de veces que son actualizados en el ordenador.

Para el usuario del ordenador esto no supone gran pérdida de información. Su pantalla se ve refrescada cada segundo, o como límite máximo cada dos segundos (nunca se alcanza este tiempo). Las variaciones en la frecuencia de refresco de la pantalla dependen del número de transmisiones que ocurren desde el sistema, y este número depende a su vez de las ramificaciones que tome el programa de la EPROM.

Por el contrario, es muy necesario que la frecuencia de toma de datos del sistema sea mayor, a fin de establecer un control más eficaz. El conjunto completo de nueve datos (corrientes en las cinco líneas, en los tres generadores y estados de los mismos) es actualizado en el sistema microprocesador cada 13 ms., por las características de programación del reloj del convertidor.

El sistema garantiza la estabilidad de la central hasta tal punto de que antes de producirse el desacople de un generador por sobrecarga se desconectarían las líneas, todas si es preciso. La situación intolerable sería que los generadores saltaran uno tras otro y se produjera un cero de potencia en la central.

Los tiempos de retardo y testeo de valores pueden ser modificados con mínimas variaciones en el software. También pueden ser regulados los tiempos de espera de estabilización de los generadores tras una descarga, actualmente ajustados a 0,5 segundos.

Los requerimientos del sistema no incluyen la carga de líneas. Esta será llevada a cabo manualmente por los operarios de la central. No obstante se puede implementar un sistema de control que la realice de forma automática, decidiendo la línea a cargar e incluso el momento adecuado para efectuarla. Los únicos cambios necesarios son de tipo software, añadiendo una nueva rutina a las ya existentes.

Estos detalles junto con los ya expuestos en la segunda parte del proyecto son suficientes para conocer con exactitud las características y limitaciones del sistema.

P R U E B A S D E M A N T E N I M I E N T O

La adjunción de pruebas de examen del sistema en un manual de mantenimiento es bastante positivo, puesto que agiliza y facilita la labor de mantenimiento y reparación.

Las pruebas que se pueden proporcionar son de varios tipos:

- Pruebas software:

Consiste en pequeños programas de autotesteo del sistema, suministrados en la misma memoria ROM donde corre el programa normal, y se encarga de hacer un chequeo de los distintos componentes que forman el sistema, o al menos de los más importantes.

Otra forma de prueba software consiste en desarrollar un programa aparte, en otra EPROM distinta, que se denominarla EPROM de prueba. El programa que corre en ella

se destinaría a comprobar las respuestas del sistema ante ciertos estímulos prefijados. Según estas respuestas el reparador sabría si el sistema funciona adecuadamente.

- Pruebas hardware:

Se trata de la verificación de ciertas señales. El fabricante suministra valores de tensión, corriente, frecuencia, niveles lógicos, ondas,... que deben aparecer en distintos puntos del sistema durante su funcionamiento normal o en cualquier instante indicado. A la hora de la reparación o mantenimiento, el usuario comprobaría la presencia de estos valores en los puntos especificados y le ayudaría a conocer la causa de la avería en caso de que ésta exista.

MEDICIONES

Veamos una relación de medidas tomadas en diversos puntos del circuito, que resultan claves en el buen funcionamiento del mismo.

1) El microprocesador 8085.-

Las señales más importantes a testear en el

microprocesador, o al menos las más sospechosas ante una avería, son las de reset y la de reloj.

Es conveniente comprobar la buena respuesta del circuito anexo a la patilla de RESET IN. En estado normal esta patilla debe permanecer a uno lógico. Al activar el pulsador la patilla mencionada va a cero, y cuando vuelve a soltarse el pulsador nuevamente queda en uno. Algunas veces esto no ocurre así, pudiendo existir un nivel alto reconocible por el sistema.

Por otra parte se precisa vigilar la entrada de reloj en caso de averías. Esta debe ser una entrada bien definida entre los valores TTL alto y bajo, cuadrada y de la frecuencia requerida. Normalmente los clocks generados a partir de circuitos RC son poco fiables. Este no es nuestro caso.

El sistema implementado dispone de un cristal de cuarzo de 6,114 MHz que proporciona una señal de reloj bastante fiable. Anomalías en la señal de reloj son fatales para el funcionamiento de todo el sistema, ya que el secuenciamiento de las instrucciones no puede realizarse convenientemente.

Una de las señales que también debe ser testeada es la

entrada de interrupción RST 6.5, en la cual tiene que aparecer una onda concreta, indicada más adelante.

Asegurarse de que el resto de las interrupciones están inhabilitadas, o conectadas a masa, para evitar posibles conflictos debido a ruidos.

2) Los integrados 8155.-

Lo más destacado en ellos a la hora de un testeo hardware son las señales de clock. Ambos integrados utilizan el timer: el primero está programado a 51 KHz para el convertidor ADC; el segundo está programado para una salida de 76 KHz destinada a la USART. Comprobar que estos valores son ciertos es conveniente.

3) El módulo de comunicación.-

En esta parte resultan de importancia ciertas señales como las de reloj CLK, TxC y RxC.

La primera coincide con la señal de reloj suministrada por el microprocesador a través de la salida CLKOUT, y debe ser de 3,057 MHz. Las otras dos están puenteadas y proceden de la salida TIMEOUT del timer del segundo

integrado 8155. Su valor debe ser de 76 KHz.

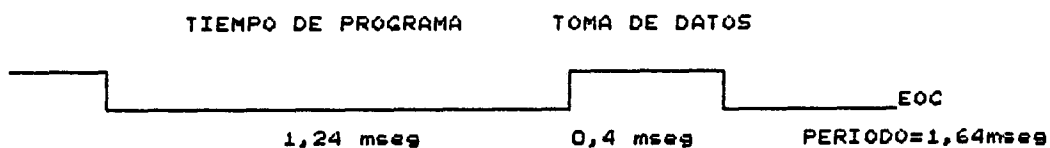
La entrada TxD y la salida RxD varían entre el nivel 1 y 0 lógicos TTL mientras se producen las comunicaciones.

Es interesante testear las entradas y salidas de estas señales a través de la interface RS232-C, proporcionada por los integrados MC1488 y MC1489. Estas deben estar comprendidas entre los valores de +12/-12 V, o un poco menos, +10/-10 V.

4) El convertidor ADC0808.-

Una señal importante a verificar es la EOC (fin de conversión). Esta patilla es la encargada de activar la interrupción RST 6:5, por lo que es fundamental que se produzca para realizar la toma de datos.

La forma de onda es la que sigue:



Durante su estado a nivel cero la conversión del dato está teniendo lugar. Es el tiempo de ejecución del programa normal. Durante su nivel alto se está realizando la lectura del dato convertido y se dan órdenes para la siguiente conversión.

Esta señal debe coincidir con la de interrupción RST 6.5 del microprocesador.

Por otra parte se debe examinar la entrada de reloj al ADC: una señal cuadrada de 51 KHz procedente de la salida del timer programable de la primera 8155, TIMEOUT.

Es necesario que la tensión de referencia para la conversión sea muy precisa, a fin de que los resultados digitales no se falseen. En esta entrada de 5 V existe un regulador de tensión, ajustado a la tensión indicada. Se dispone de un potenciómetro de ajuste que puede variarse en caso de que se detecte una tensión distinta de 5 voltios en la entrada de referencia.

5) Entradas analógicas.-

Las entradas analógicas son corrientes comprendidas entre 4 y 20 mA, en la situación real. No deben superarse los 20 mA de intensidad.

En principio, los 20 mA convertidos a través del conversor corriente-tensión tienen que proporcionar la tensión de 5 V, a la salida de los amplificadores operacionales. Esto supondría también el límite máximo para el convertidor A/D, el cual 5 V los transforma en 255 decimal (la cifra límite que puede representar).

Cualquier exceso de corriente por encima de los 20 mA produciría una tensión mayor de 5 V en la entrada analógica del convertidor A/D. Tal exceso no puede ser soportado por mucho tiempo, pues el ADC0808 está diseñado para una tensión analógica de entrada de un máximo de 5 V.

A modo de protección ante la aparición de una corriente mayor de 20 mA se ha diseñado el conversor corriente-tensión de forma que 20 mA correspondan a 4,4 V. Es en definitiva un modo de protección del convertidor ADC, pudiendo tolerarse entradas de hasta 22 mA.

6) Los optoacopladores.-

Para verificarlos se inyecta a la entrada del circuito anexo a ellos una tensión de cinco a cero voltios. La salida del circuito coincide con la entrada al puerto A del segundo 8155.

Una tensión de entrada al optoacoplador de 5 V se detecta a la salida (puerto A) como un cero lógico. Y viceversa, una entrada de cero voltios se detecta en el puerto como un uno lógico.

No se debe aplicar a la entrada del optoacoplador más de 5 V. No obstante, la circuitería dispone de un diodo zener de 6,2 V que limitará la tensión de entrada. Tampoco es reconocida una entrada de 4 V, por lo que habrá que aplicar tensiones mayores para conseguir excitar al optoacoplador. Además, la corriente por el diodo de entrada al optoaislador no debe superar los 80 mA. Tal como se ha diseñado la circuitería anexa, esta corriente en nuestro caso no supera los 60 mA.

7) Los relés. Etapa de potencia.-

El puerto B de salida (segundo 8155) es el correspondiente a los relés, conmutados a través de una etapa de potencia. Los relés en miniatura utilizados tienen un número de actuaciones limitado, por lo que en caso de que se sospeche que alguno falla habrá que comprobarlo, aplicándole una tensión de 12 V entre sus terminales, teniendo en cuenta además que su resistencia interna es de 280 óhmios.

La no conmutación de un relé también podría ser causada por el fallo del transistor del circuito amplificador, o fallo en la salida del puerto, produciendo una pésima excitación del transistor.

En general, las señales anteriormente citadas nos pueden indicar si el programa del sistema se está ejecutando convenientemente. En caso de que una presente cierto comportamiento anormal tendremos un indicio de fallo y nos resultará relativamente fácil averiguar la fuente del mismo.

Señales como RD, WR, ALE, son utilizadas por varios circuitos del sistema. Es aconsejable comprobar que estos circuitos las reciben, además de activarse sus correspondientes "chip enable" (las salidas del decodificador de direcciones 74138, o su equivalente 8205). La no activación de tales señales impediría al microprocesador acceder a los distintos integrados.

En general, una señal TTL se considerará nivel alto con 3,5 V y nivel bajo con 0,2 V aún cuando su histéresis es mayor. Para mejor información consúltense las características de los integrados adjuntas.

Las potencias de consumo del sistema deben vigilarse para cerciorarnos de que todo va bien. Un consumo excesivo en relación con el habitual, o todo lo contrario, un consumo demasiado pobre, son claros signos de que algo falla.

Antes de comenzar la ejecución del programa, la relación de consumos es la siguiente:

* fuente de -12V --- 0,12 A

* fuente de +12V --- 0,15 A

Y en funcionamiento normal, es decir, con todas las cargas conectadas, se tiene:

* fuente de -12V --- 0,12 A

* fuente de +12V --- 0,50 A

* fuente de +5V --- 0,40 A

AUTOTESTEO DEL SISTEMA

En la sección de "Localización de averías en sistemas con microprocesador" se expuso la forma de realizar distintos testeos en un sistema microprocesador: testeo

del microprocesador, de RAM, de ROM, de puertos, del convertidor. Todos estos aspectos son aplicables a nuestro sistema, aunque algunos, como el testeo del convertidor A/D, requieren una alteración del hardware.

Se ha implementado un pequeño programa de verificación de la EPROM. Se basa en el sistema de checksum, es es, realización de la suma de todos los bytes de memoria, mediante una OR exclusiva, y posterior comparación del resultado con uno ya existente en memoria. Si ambos datos son distintos es prueba de que la EPROM no está en buenas condiciones y será necesario reemplazarla.

Asimismo resulta sencillo realizar un testeo del microprocesador, simplemente haciéndole ejecutar un par de instrucciones y comprobando que los resultados son los correctos.

Un testeo de RAM también es fácil; simplemente se escribe un dato en ella, se lee y se compara ambos. A continuación se lista el programa de checksum.

ASMB00 TEST.ASM PAGEWIDTH(60) PAGELENGTH(54)

ISIS-II 8080/8085 MACRO ASSEMBLER, V4.0

MODULE PAGE 1

LOC	OBJ	LINE	SOURCE STATEMENT
		1	PUBLIC TEST
		2	EXTRN TESTEO
		3	CSEG
		4	TEST:
0000	3E00	5	MVI A,00
0002	F5	6	PUSH PSW
0003	2600	7	MVI H,00H
0005	2E00	8	MVI L,00H
0007	160D	9	MVI D,0DH
0009	1E00	10	MVI E,00H
000B	F1	11	AO: POP PSW
000C	AE	12	XRA M
000D	23	13	INX H
000E	F5	14	PUSH PSW
000F	7A	15	MOV A,D
0010	BC	16	CMP H
0011	C20B00	17	JNZ AO
0014	7B	18	MOV A,E
0015	BD	19	CMP L
0016	C20B00	20	JNZ AO
0019	F1	21	POP PSW
		22	
001A	320000	23	STA TESTEO
001D	C9	24	RI10: RET
		25	END

PUBLIC SYMBOLS
TEST C 0000

EXTERNAL SYMBOLS
TESTEO E 0000

USER SYMBOLS
AO C 000B RI10 C 001D TEST C 0000 TESTEO E 0000

ASSEMBLY COMPLETE, NO ERRORS

P O S I B L E S A M P L I A C I O N E S

Y M E J O R A S

Las posibles ampliaciones del sistema se han ido nombrando poco a poco durante todo el desarrollo del sistema.

La ampliación más importante a la que nos podemos referir es a la del aumento en el número de líneas y generadores de la central a controlar. Esto implicaría pocos cambios de tipo software, pero variaciones considerables en el hardware: aumento del número de puertos, disposición de nuevos convertidores analógico-digitales, circuitería anexa a la toma de datos y envío de órdenes,...

En cuanto a las mejoras del sistema, podemos recurrir a otros componentes. La tecnología avanza tanto y tan

rápidamente que todo los días surgen nuevos elementos, nuevas herramientas y métodos optimizados para cualquier tipo de control. Aparte de las mejoras por sofisticación de los materiales utilizados, están las mejoras por adición de nuevas posibilidades de control. Trataremos de enumerar algunas, aunque insistimos, son tantas como imaginación tenga el diseñador.

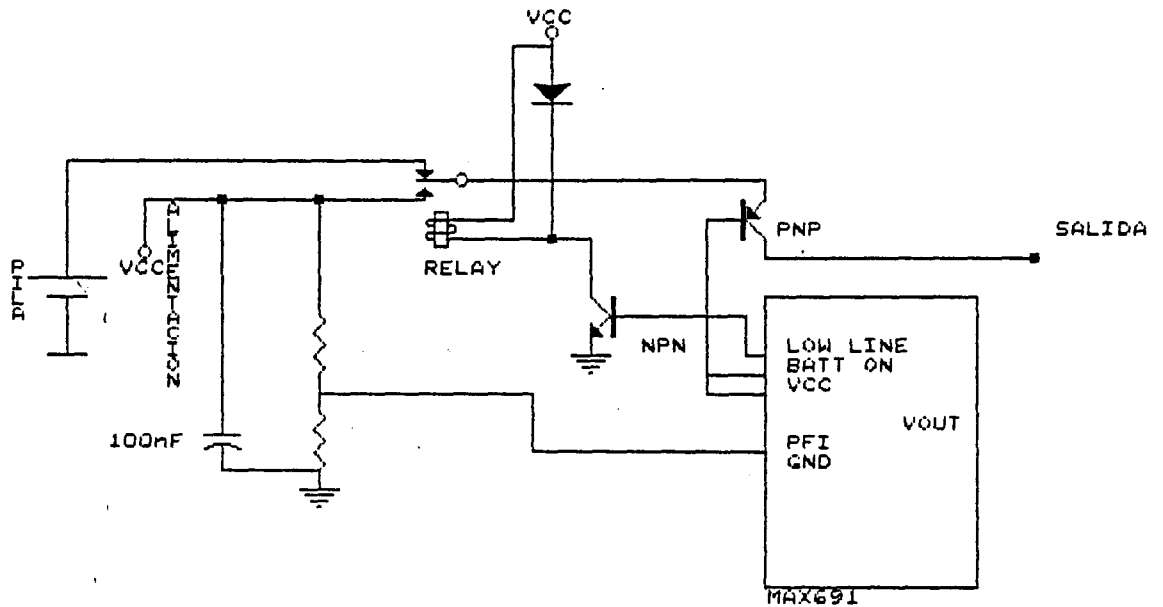
MEJORAS EN EL MANTENIMIENTO

Citemos algunas mejoras posibles en cuestión de mantenimiento del sistema:

* La duplicación del equipo, a fin de prevenir la avería en alguna tarjeta. En caso de fallo en el sistema, éste realizarla inmediatamente la conmutación de tarjetas, mediante una señal de aviso.

* Tener una pila preventiva en caso de que el sistema quede sin alimentación. Al ocurrir un fallo en la alimentación la pila entraría en funcionamiento, alimentando al sistema.

El esquema para la implementación de esta pila podría ser el de la figura siguiente:



En paralelo con el relé es conveniente la colocación de un integrado CMOS de elevada potencia que detectase rápidamente la señal de LOW LINE y fuera quien actuase en principio. El relé, por tener un tiempo de respuesta mayor conmutaría más tarde. Con este integrado en paralelo con el relé evitáramos que por un tiempo de respuesta grande, los integrados del sistema perdieran la información y posiblemente se bloqueara el sistema, o le hiciera tomar decisiones erróneas.

* Añadir al sistema la circuitería necesaria para realizar un testeo del convertidor. La colocación de unos

microswitchs en las entradas analógicas harían posible la conmutación, entre las señales reales correspondientes a las corrientes de los elementos y otras señales conocidas de antemano. Los valores de estas últimas estarían grabadas en la EPROM, y se comprobaría si al realizar la conversión se obtienen efectivamente los valores esperados.

* Utilizar en la comunicación integrados que presenten un gran rechazo al modo común. Con esto conseguimos que si se crea un campo magnético de perturbación alrededor del cable de comunicación, tal que hiciese subir el nivel de tensión de la línea, se rechazase este ruido, quedándose sólo con la entrada diferencial.

Como ejemplo de estos integrados están: HS-245 (transmisión) y el HS-246 o HS-249 (recepción), cuyas características se adjuntan al final.

MEJORAS EN CUANTO A CONTROL

El software puede ser ampliado con nuevos detalles que enriqueciesen las posibilidades del sistema.

*, Después del descargo de una línea se tardan unos 100

mseg. en reconocer que efectivamente ha ocurrido esa descarga. El sistema podría testear tras ese tiempo si la descarga ha ocurrido. Si no ha sido así, se procedería a la emisión de un mensaje de error al ordenador, indicando tal averencia, además de especificar en qué línea concreta sucedió el error.

* Si en algún momento la suma de los consumos es mayor que la suma de la generación, posiblemente se esté ante un caso de cortocircuito en alguna de las líneas. También se podría visualizar ese fallo en el ordenador, tratando de llamar la atención del usuario.

* Se podría idear la forma de detectar el caso de que un generador se convierta en motor, e indicarlo asimismo en el ordenador.

* Sacar un parte por impresora cada vez que ocurra una situación anómala.

* La visualización de datos en el ordenador sería factible realizarla en dos pantallas: una con los valores de líneas y generadores, y otra con los mensajes de error.

* La implementación de un software destinado a la comprobación del sistema resulta enormemente útil. En una

EPROM podría grabarse un programa de verificación, que por sí solo generase los valores necesarios y así el operario podría observar las respuestas obtenidas.

S I M U L A C I O N E N T I E M P O R E A L

Todo desarrollo de circuitos electrónicos se divide en dos fases: diseño del circuito y verificación de su funcionamiento.

La verificación del funcionamiento puede realizarse mediante los métodos tradicionales (empleo de osciloscopio, analizadores lógicos, generadores de señales), o bien mediante métodos informáticos (CAE).

Para la verificación de nuestro sistema se ha empleado el emulador del sistema de desarrollo MDS-221 de Intel. Se ha realizado con él una simulación en tiempo real. Es decir, mediante programas del sistema de desarrollo, interrupciones del mismo y su emulador, sumado al software y hardware del prototipo a probar, se ha verificado el comportamiento del sistema de control. Esta verificación tiene la características de ser en tiempo real, o sea, con

los mismos requerimientos de tiempo que en la práctica.

De esta forma ha sido posible provocar varias situaciones y examinar los resultados: se somete al prototipo a distintas situaciones posibles en la realidad, se le hace trabajar y se estudia las distintas respuestas del mismo ante los diferentes valores aplicados. El objetivo es obtener los resultados esperados.

Dos de los métodos para trabajar en procesamiento en tiempo real son:

- el uso de interrupciones.
- procesamiento paralelo.

Para el uso de interrupciones, en principio hay que tener claro que el programa de usuario que acceda al servicio de interrupciones no deberá interferir en las operaciones de alta velocidad de los dispositivos de entrada/salida. Para ello es aconsejable que hayan procesadores esclavos.

En algunos casos es necesario avisar al sistema de algún evento, pero no es necesario un requerimiento del master, de ahí la necesidad de utilizar procesadores esclavos que retengan las peticiones de interrupción. Un ejemplo claro sería un sistema de entrada/salida serie o

paralelo, en el que interactuarían un procesador esclavo y el dispositivo de entrada/salida. Este procesador esclavo trabajarían con el dispositivo de entrada/salida mientras el maestro no solicite otra petición de mayor prioridad.

El procesamiento paralelo se basa en que un número de procesadores están trabajando en paralelo, implicando un proceso más rápido. Para el procesamiento paralelo hay que tener en cuenta que los algoritmos reduzcan al mínimo la falta de paralelismo, y que no hayan grandes diferencias de niveles entre procesos paralelos, ya que si no habría momentos en que una parte de los procesadores estarían en estado de espera.

La emulación en tiempo real permite modificar el programa, pararlo y hacer mil cosas con él hasta haber visto todas las situaciones posibles.

La emulación en tiempo real implica que el procesador de la tarjeta emuladora ejecuta las instrucciones del usuario, que pueden estar cargadas tanto en dicha tarjeta como en el montaje práctico. Las operaciones del sistema podrán ser controladas bajo el monitor del emulador. La emulación en tiempo real permite que el emulador pueda hacer correr el programa desde el principio hasta que se den unas condiciones de software o hardware, con las que

se parará la emulación y se retornará el control a la consola.

En el caso del MDS-221 los comandos y posibilidades para una emulación en tiempo real son:

Para inicializar la emulación lo primero a realizar es el mapeado de memoria, definiendo lo que se deja al ICE y lo que va a utilizar el montaje de prueba.

Con el comando GO el programa comenzará su ejecución. Si no se ha definido un contador de programa, éste partirá de la primera instrucción ejecutable del programa. En este caso sólo se podría parar el programa con un ESCAPE o si el programa hiciese un acceso a posiciones de memoria no definidas o protegidas.

Con el comando GO también se puede forzar un punto de parada. Por ejemplo:

```
GO FROM (dirección inicial) TILL (dirección final)
```

Las direcciones pueden ser dadas por un valor determinado o por el nombre de una etiqueta. Con ello podremos emular una parte del montaje, detectando así posibles errores.

El emulador proporciona un par de registros, BR0 y BR1 para crear una parada en el sistema. Su configuración viene dada por la siguiente tabla.

System Group Name	Channel-numbers	8085 Pin or Signal	Interpretation	Trace Radix
U0	1 - 8		User probe channels	H
U1	9 - 16		User probe channels	H
U2	17, 18		User probe channels	H
DMUX	19	ALE	1 = AD0 to AD7 are data 0 = AD0 to AD7 are low-order address	Y
ADDR	20 - 35	AD0 - AD7 A8 - A15	Low-order address lines (DMUX=0) High-order address lines	H
DATA	20 - 27	AD0 - AD7	Data Lines (DMUX=1)	H
ADDRL	20 - 27	AD0 - AD7	Low-order address lines (DMUX=0)	H
ADDRH	28 - 35	A8 - A15	High-order address lines	H
STS	36 37 38	S0 S1 IO/M	Action IO/M S1 S0 Mnemonic Mnemonic HALT 0 0 0 H WRITTEN 0 0 1 W READ 0 1 0 R EXECUTED 0 1 1 E OUTPUT 1 0 1 O INPUT 1 1 0 I	
SD	39 40	SOD SID	Serial output data line Serial input data line	Y
RW	41 42	WR RD	WR line RD line	Y
MTH	43	MATCH 0 or MATCH 1	1 = breakpoint register matched in previous frame (trace data only)	Y

Para la programación de estos registros se utiliza el registro BR. El comando de habilitación de los mismos es GR. Este comando habilita un punto de ruptura definido por estos registros. Cada uno de ellos consta de 43 bits. Los 18 primeros corresponden a canales de prueba del usuario, los siguientes 24 corresponden a señales del 8085, y el último es el indicador de que ha habido una ruptura, por coincidir el registro programado con el que el emulador estudiaba en ese momento. En cada ciclo el emulador compara el registro BR con CSR (registro de estado del canal).

El CSR se actualiza en cada flanco. Hay dos flancos por instrucción: el primero se activa cuando hay direcciones en el bus de datos, y el segundo cuando hay datos. De cualquier forma, el comando CAUSE permite al usuario conocer cuál ha sido la causa de parada. La tabla siguiente indica cómo saberlo, basándose en los bits puestos a uno:

Status-register ICE-85 Status Register and Interpretation

OPCODE	Opcode fetched in last opcode-fetch cycle in trace data (8 bits, read-only)
CAUSE	Conditions that were true at the time of the last break in emulation (read-only). The 8 bit values are: Bit 0 on if breakpoint 0 matched Bit 1 on if breakpoint 1 matched Bit 2 on if SY0 caused break Bit 3 on if guarded memory or I/O accessed Bit 4 on if user aborted processing Bit 5 on if timeout on HOLD Bit 6 on if timeout on READY Bit 7 on if no user Vcc (power supply voltage)
PPC	Previous program counter; address of last instruction-fetch cycle in trace data (16 bits, read-only).
PSW	Program status word; accumulator in low byte, status flag register in high byte (16 bits, read-only).
UPPER	Highest location in Inteltec memory available to map user memory into (16 bits, read-only).
BUFFERSIZE	Number of frames of valid trace data; initially zero, always between zero and 1022 (16 bits, read-only).
TIMER	Low 16 bits of 2-MHz emulation timer.
HTIMER	High 16 bits of timer.

También se pueden realizar paradas mediante las interrupciones SY0IN y SY1IN. Actualmente el sistema de desarrollo de que disponemos no posee los conectores que permitirían este tipo de conexiones hardware.

En combinación con un registro de parada, el comando GR puede tener además una dirección de parada. Por ejemplo:

Definir BR0 (o BR1)
GO FROM 3000 read TILL BR0

En la simulación en tiempo real se consideran diferentes ficheros, y mientras corre el emulador con el programa de estabilidad, hay un testeo de si se ha pulsado algún número indicativo de la situación que queremos simular.

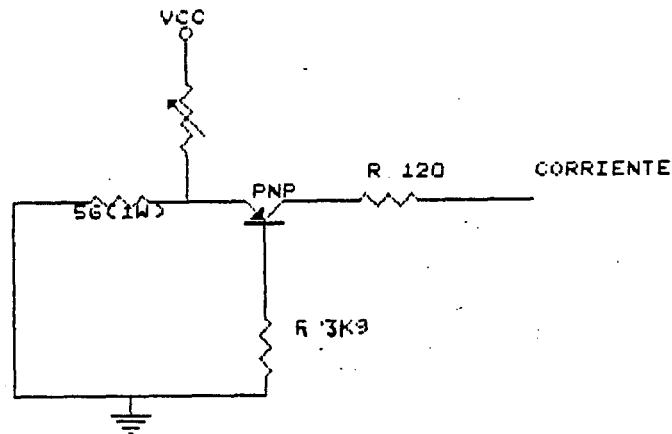
PROGRAMAS ADAPTADOS PARA LA SIMULACION

A través del sistema de desarrollo MDS-221 se consigue la simulación del sistema microprocesador, tal como se ha indicado. Los programas ejecutables de esta versión de simulación son básicamente los grabados en una EPROM (versión MODIF3), pero con ciertas modificaciones necesarias para la simulación.

Así se prescinde de la simulación de líneas y generadores que se habla utilizado para el programa de EPROM, y que consistía en ocho generadores de corriente aplicados a las entradas analógicas del sistema. Estos generadores implementados ofrecían demasiadas

limitaciones a la hora de provocar todas las condiciones de funcionamiento posibles.

Los ocho generadores fueron cableados sobre una misma placa de prueba, y su estructura es la siguiente:



Los potenciómetros de que disponen permiten variar las corrientes dentro de un margen que va desde 10 mA a 22 mA. Estar todas las fuentes de corriente montadas sobre la misma placa trae los inconvenientes de que se influyen unos a otros, provocando corrientes erróneas. Es decir, al variar el potenciómetro de uno de los generadores las corrientes varían no sólo en el generador correspondiente, sino en algún otro.

Aparte de esto, el conjunto de las fuentes no cumple

las características de una situación real en una central eléctrica, tales como son:

- La suma de las corrientes de los generadores debe ser igual a la suma de las corrientes en las líneas, de tal forma que la generación sea igual a la demanda. No obstante puede existir una pequeña diferencia debido al consumo de los servicios auxiliares.
- Si una de las líneas es descargada, la potencia en los generadores debe disminuir según la potencia que consumía la línea antes de producirse el descargo.
- De la misma forma, si la potencia en una o varias líneas disminuye también disminuirá la potencia en la generación.
- Si un generador se desacopla, los restantes aumentarán su generación en la misma magnitud, de forma que darán la potencia que antes era suministrada por el generador desacoplado además de la que ya suministraban.

Estas peculiaridades, indudablemente, no pueden obtenerse de los generadores de corriente implementados.

Por todo esto se ha dicho en repetidas ocasiones que no es posible una simulación eficiente si no es a través del propio sistema de desarrollo.

La diferencia radical entre el programa grabado en EPROM y el de simulación es que en este último no se produce una toma de datos real.

El programa de la EPROM recibe una interrupción periódicamente que utiliza para recoger el dato convertido por el ADC y activar la siguiente conversión. El programa de simulación, al que se ha llamado MODIFW, también recibe una interrupción periódica RST 6.5, pero realizar una falsa toma de datos, de modo que los valores de corriente no se los proporciona las fuentes de corriente, sino el propio programa de simulación, haciendo trabajar al sistema con unos valores de potencias previamente establecidos por software. Los datos sobre las potencias de líneas y generadores son programados adecuadamente para provocar distintas situaciones.

El programa MODIFW proporciona ocho situaciones diferentes de simulación, definidas por ocho conjuntos de datos. Se utiliza una variable SI que puede tomar valores enteros entre 0 y 7, y según sea su valor el sistema tomará un paquete de datos distinto y se correrá el

programa como si esos fuesen los datos reales.

La ejecución de cualquiera de estas simulaciones se elige a través del teclado. El programa de simulación está preparado para leer los datos introducidos por él.

Al final del bucle de estabilidad el programa testea si ha sido pulsada una tecla. En tal caso se lee. La tecla pulsada puede estar comprendida entre el valor 0 y el 7. El valor es recogido por el programa desde el buffer de teclado y se almacena en la variable SI. Cuando el bucle de estabilidad vuelve al principio se testea el contenido de la variable, y según cual sea el valor almacenado en la misma el programa procederá a la ejecución de una de las ocho posibles simulaciones.

El paso de una simulación a otra se efectúa fácilmente. Es conveniente regresar al punto 0 (simulación inicial) después de cada simulación y antes de comenzar una distinta.

Condiciones de simulación.-

Para cada valor de la variable SI, como se ha dicho, se tiene una simulación distinta. Veamos cuáles son:

Simulación inicial.-

En este caso la variable SI toma el valor cero. Se presenta una situación de normalidad, esto es, estable. Todos los grupos están acoplados y todas las cargas conectadas. Las potencias de cada elemento se mantienen dentro de los márgenes normales, por debajo o igual a sus potencias nominales. Las respuestas del sistema se limitan a permanecer en el bucle de estabilidad indefinidamente, vigilando que los valores de potencias no sobrepasen los límites establecidos.

Simulación número 1.-

Este es el caso en que la variable SI toma el valor 1. Se simula la situación en la que un generador se sobrecarga. Como consecuencia se entra en la rutina de testeo, se asigna un tiempo de testeo que como máximo será de 2,5 segundos, y tras este tiempo se volverá a consultar si el generador en mal estado se recupera. Efectivamente lo hace, y no es necesario efectuar descarga alguna.

Simulación número 2.-

La variable SI toma el valor 2. El grupo I adquiere una

sobrecarga mayor del 10%. Se entra en la rutina de testeo y se espera cierto tiempo. Tras él se vuelve a comprobar el valor del generador que estaba mal y éste no ha vuelto a la normalidad, por lo que se procede a la descarga de una línea. La línea a desconectar es elegida mediante el criterio siguiente: se desconecta aquella cuyo valor de potencia sea el 10% de la suma de los generadores, o por encima de este valor.

Después de la descarga, el generador que sufría un exceso se estabiliza y el sistema recupera su estabilidad.

Simulación número 3.-

En ella la variable SI toma el valor 3. Su actuación es idéntica a la anterior, con la única diferencia de que el generador que presenta la anomalía es el número 2.

Simulación número 4.-

El valor de SI es 4, y el funcionamiento es análogo al anterior, con la única diferencia de que es el grupo III el que sufre la sobrecarga.

Simulación número 5.-

El valor de la variable SI es 5. En este caso uno de los grupos se desacopla, por cualquier avería. Los otros dos generadores lógicamente se sobrecargan, pues tienen que suministrar aquella potencia que antes daba el desacoplado. Como consecuencia de ello se sobrecargan y es necesario una descarga. Tras la desconexión de una línea el sistema se estabiliza. El grupo que se desacopla es el número 1.

Simulación número 6.-

El valor de SI es 6. La situación que se plantea es análoga a la anterior, con el cambio de que en esta el generador desacoplado es el número 2.

Simulación número 7.-

Esta es la última simulación. El valor de SI es 7, y la situación es semejante a las dos anteriores, con la diferencia de que es el grupo III el que se desacopla de la red.

Para concluir queda decir que la simulación en tiempo real llevada a cabo tiene varios objetivos:

- El primero de todos ha sido la verificación del circuito implementado, y una buena herramienta de ayuda para la depuración.

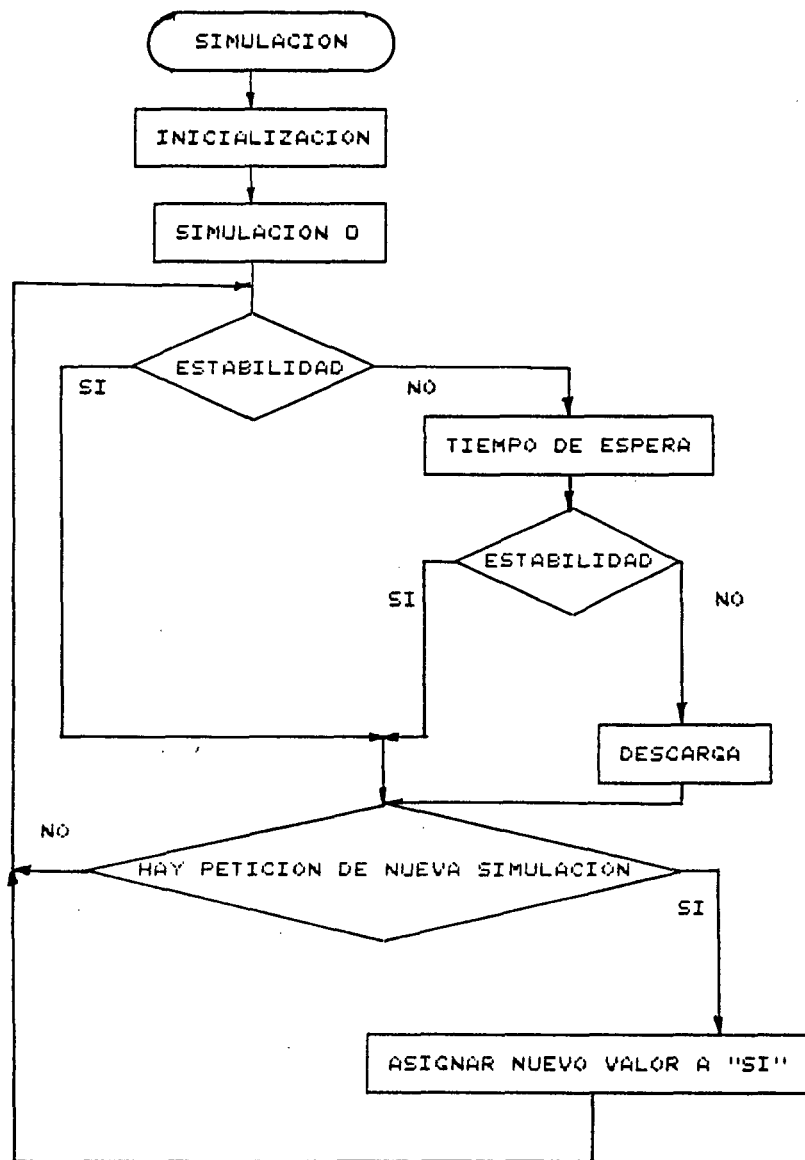
- El segundo objetivo es proporcionar unos estímulos similares en todo lo posible a los reales, para realizar la presentación del mismo al cliente o futuro usuario, mostrándole sus capacidades y limitaciones.

- El tercer objetivo es confeccionar un programa que, aparte de los puntos mencionados, sirva para la reparación del sistema en cualquier momento. Es decir, las versiones de simulación corridas en el sistema de desarrollo y el emulador de éste son instrumentos útiles en cualquier momento de la vida del aparato de control, cuando se requiera un examen del mismo, bien para verificar su buen funcionamiento (a nivel de mantenimiento), o para ayuda en la reparación.

La grabación en una EPROM de este programa de simulación proporcionaría al usuario un medio de

verificación del sistema, con sólo sustituir la EPROM normal por la de verificación, a la hora de llevar a cabo ésta.

ORGANIGRAMA DE SIMULACION



P R O G R A M A S

V E R S I O N M O D I F W D E

S I M U L A C I O N

ISIS-II PL/M-80 V3.1 COMPILATION OF MODULE ESTABILIDAD
 OBJECT MODULE PLACED IN MODIFW.OBJ
 COMPILER INVOKED BY: PLM80 MODIFW.PLM WORKFILES(:F0:,:F0:)
 -DEBUG PAGELLENGTH(54) PAGESWIDTH(60)

```

/* Programa principal de control de es
-  tabilidad */
1      ESTABILIDAD:DO;

2      1      DECLARE IG(3) BYTE PUBLIC;
3      1      DECLARE IL(5) BYTE PUBLIC;
4      1      DECLARE FLG(3) BYTE PUBLIC;
5      1      DECLARE (LINDESCONECTADAS,CANALDAC,INIC,CC
-  R,SI) BYTE PUBLIC;
6      1      DECLARE (ALFA,BETA,INOMINAL) BYTE PUBLIC;
7      1      DECLARE COMIEN LABEL PUBLIC;

8      1      TRANSMISION$AL$PC:PROCEDURE PUBLIC;
9      2      DECLARE USARDATOS LITERALLY '10H';
10     2      DECLARE USARCONTROL LITERALLY '90H';
11     2      DECLARE PORTB$RAM1 LITERALLY '42H';
12     2      DO;
13     3      LAZ04: B=INPUT(USARCONTROL) AND 02
-  H;
14     3      IF B<>02H THEN GOTO LAZ04;
16     3      B=INPUT(USARDATOS);
17     3      IF B='#' THEN RETURN;
19     3      IF B<>'4' THEN GOTO LAZ04;
21     3      LAZ06: B=INPUT(USARCONTROL) AN
-  D 01H;
22     3      IF B=0 THEN GOTO LAZ06;
24     3      OUTPUT(USARDATOS)='4';

25     3      DO I=0 TO 2;
26     4      LAZ01: B=INPUT(USARCONTROL) AND
-  01H;
27     4      IF B=0 THEN GOTO LAZ01;
29     4      OUTPUT(USARDATOS)=IG(I);
30     4      END;
31     3      DO I=0 TO 4;
32     4      LAZ02: B=INPUT(USARCONTROL) AND
-  01H;
33     4      IF B=0 THEN GOTO LAZ02;
35     4      OUTPUT(USARDATOS)=IL(I);
36     4      END;
37     3      LAZ03: B=INPUT(USARCONTROL) AND
-  01H;
38     3      IF B=0 THEN GOTO LAZ03;
40     3      OUTPUT(USARDATOS)=LINDESCO

```

```

-   NECTADAS;
41   3   TRI0:           RETURN;
42   3           END;
43   2           END TRANSMISION$AL$PC;

44   1   PRU1:PROCEDURE EXTERNAL;
45   2   END PRU1;

46   1   RUTTESTEO:PROCEDURE (DIRPI) EXTERNAL;
47   2           DECLARE DIRPI ADDRESS;
48   2           END RUTTESTEO;

49   1   S$MASK: PROCEDURE (MASK) EXTERNAL;
50   2           DECLARE MASK BYTE;
51   2   END S$MASK;

52   1           DECLARE CONTROL$RAM1 LITERALLY '40H';
53   1           DECLARE USARCONTROL LITERALLY '90H';
54   1           DECLARE USARDATOS LITERALLY '10H';
55   1           DECLARE CONTROL$RAM2 LITERALLY '30H';
56   1           DECLARE PORTA$RAM2 LITERALLY '31H';
57   1           DECLARE PORTB$RAM2 LITERALLY '32H';
58   1           DECLARE PORTA$RAM1 LITERALLY '41H';
59   1           DECLARE PORTB$RAM1 LITERALLY '42H';
60   1           DECLARE TIMERL$RAM2 LITERALLY '34H';
61   1           DECLARE TIMERH$RAM2 LITERALLY '35H';
62   1           DECLARE TIMERL$RAM1 LITERALLY '44H';
63   1           DECLARE TIMERH$RAM1 LITERALLY '45H';
64   1           DECLARE IN LITERALLY '212';
65   1           DECLARE P(3) BYTE;
66   1           DECLARE (W,B,I,ED) BYTE;
67   1           DECLARE ADC(8) BYTE AT (2000H);

68   1           TOMA$INICIAL: PROCEDURE;
69   2           DO;
70   3               INIC=0FFH;
71   3               CANALDAC=00;
72   3               ADC(CANALDAC)=00;
73   3               ENABLE;
74   3           CUCO: IF INIC=0 THEN DO;
76   4               LINDESCONECTADAS=1FH;
77   4               OUTPUT(PORTB$RAM2)=LINDESCONEC

-   TADAS;
78   4           RETURN;
79   4           END;
80   3           TI10: GOTO CUCO;
81   3           END;
82   2           END TOMA$INICIAL;

```

```

63 1 PCALFABETA: PROCEDURE;
64 2 DO;
65 3 BUCLE3: B=INPUT(USARCONTROL) AND 01;
66 3 IF B=0 THEN GOTO BUCLE3;
68 3 OUTPUT(USARDATOS)='%' ;

89 3 DO I=1 TO 40;
90 4 CALL TIME(200);
91 4 END;
92 3 BUCLE4: B=INPUT(USARCONTROL) AND 02;
93 3 IF B=0 THEN GOTO BUCLE3;
95 3 ED=INPUT(USARDATOS);
96 3 IF ED<>'%' THEN GOTO BUCLE3;
98 3 DO I=0 TO 2; /* Recibo del PC a
- lfa y beta e INOMINAL */
99 4 BUCLE2: B=INPUT(USARCONTROL) AND 01H;
100 4 IF B=0 THEN GOTO BUCLE2;
102 4 OUTPUT(USARDATOS)='&';
103 4 BUCLE: B=INPUT(USARCONTROL) AND 02H;
104 4 IF B=02 THEN GOTO ENTRADA;
106 4 GOTO BUCLE;
107 4 ENTRADA: IF I=0 THEN ALFA=INPUT(USARDATOS);
109 4 IF I=1 THEN BETA=INPUT(USARDATOS);
111 4 IF I=2 THEN INOMINAL=INPUT(USARDATOS);
113 4 END;
114 3 LAZO: B=INPUT(USARCONTROL) AND 01H; /
- * Mando confirmacion al PC */
115 3 IF B=0 THEN GOTO LAZO; /* que lo
- s datos han sido */
117 3 OUTPUT(USARDATOS)='&'; /* recibi
- dos. */
118 3 PCALFABETA: RETURN;

120 1 END PCALFABETA;

/* Fase de inicializacion del sistema */

121 1 COMIEN: DO;
122 2 OUTPUT(TIMERH$RAM1)=40H; /* Programacio
- n del klok del convertidor */
123 2 OUTPUT(TIMERL$RAM1)=03CH; /* a 51 kHz
- aproximadamente */
124 2 OUTPUT(CONTROL$RAM1)=0C0H; /* Program
- acion de la USART en asincrono */

```

```

125 2          OUTPUT(USARCONTROL)=0CFH; /*a 1200 bau
-      dios, sin paridad 4 con dos*/
126 2          OUTPUT(TINERH$RAM2)=40H; /*bits de STO
-      P*/
127 2          OUTPUT(TIMERL$RAM2)=2BH;
128 2          OUTPUT(CONTROL$RAM2)=0C0H;
129 2          OUTPUT(USARCONTROL)=07H;
130 2          OUTPUT(USARCONTROL)=37H;
131 2          OUTPUT(CONTROL$RAM1)=02H; /*PORT B e
-      ara activar LED*/
132 2          OUTPUT(CONTROL$RAM2)=02H;
133 2          OUTPUT(PORTB$RAM1)=00H;
134 2      MO1:   DO I=0 TO 2;
135 3          P(I)=0;
136 3          FLG(I)=0;
137 3          COR=0;
138 3          SI=0;
139 3          END;

          /* Compruebo si el PC esta conectado */
140 2      B=INPUT(PORTA$RAM1) AND 01H;
141 2      IF B=01H THEN DO;
142 3          ALFA=10;
143 3          BETA=10;
144 3          INOMINAL=212;
145 3          GOTO CONEX1;
146 3          GOTO CONEX1;
147 3          END;

148 2          CALL PCALFABETA;

149 2      CONEX1: CALL S$MASK(01DH); /* Programa masc
-      ara de interrupciones 6.5 */
150 2          OUTPUT(PORTB$RAM1)=01H;
151 2          CALL TOMA$INICIAL; /* Hago una toma in
-      icial de datos */
152 2          OUTPUT(PORTB$RAM1)=02H;

153 2      PRINCIPIO: OUTPUT(PORTB$RAM2)=LINDESCONECTADAS
-      ; /*Actualizo relees*/
154 2          COR=0;
155 2      PRIN2:   B=INPUT(PORTA$RAM1) AND 01H;
156 2          DO CASE SI ;
157 3          DO;
158 4          OUTPUT(PORTB$RAM2)=1FH;
159 4          IG(0)=175;
160 4          IG(1)=152;
161 4          IG(2)=175;
162 4          IL(0)=135;
163 4          IL(1)=124;

```

```
164 4 IL(2)=071;
165 4 IL(3)=105;
166 4 IL(4)=127;
167 4 END;
168 3 DO;
169 4 OUTPUT(PORTB$RAM2)=1FH;
170 4 IG(0)=231;
171 4 IG(1)=210;
172 4 IG(2)=210;
173 4 IL(0)=148;
174 4 IL(1)=105;
175 4 IL(2)=205;
176 4 IL(3)=125;
177 4 IL(4)=115;
178 4 END;
179 3 DO;
180 4 OUTPUT(PORTB$RAM2)=1FH;
181 4 IG(0)=231;
182 4 IG(1)=210;
183 4 IG(2)=210;
184 4 IL(0)=148;
185 4 IL(1)=105;
186 4 IL(2)=205;
187 4 IL(3)=125;
188 4 IL(4)=115;
189 4 END;
190 3 DO;
191 4 OUTPUT(PORTB$RAM2)=1FH;
192 4 IG(0)=210;
193 4 IG(1)=231;
194 4 IG(2)=210;
195 4 IL(0)=148;
196 4 IL(1)=105;
197 4 IL(2)=205;
198 4 IL(3)=125;
199 4 IL(4)=115;
200 4 END;
201 3 DO;
202 4 OUTPUT(PORTB$RAM2)=1FH;
203 4 IG(0)=210; 925
204 4 IG(1)=210;
205 4 IG(2)=231;
206 4 IL(0)=148;
207 4 IL(1)=105;
208 4 IL(2)=205;
209 4 IL(3)=125;
210 4 IL(4)=115;
211 4 END;
212 3 DO;
213 4 OUTPUT(PORTB$RAM2)=1FH;
214 4 IG(0)=045;
```



```

215 4          IG(1)=230;
216 4          IG(2)=230;
217 4          IL(0)=135;
218 4          IL(1)=124;
219 4          IL(2)=071;
220 4          IL(3)=105;
221 4          IL(4)=127;
222 4          END;
223 3          DO;
224 4          OUTPUT(PORTB$RAM2)=1FH;
225 4          IG(0)=230;
226 4          IG(1)=045;
227 4          IG(2)=230;
228 4          IL(0)=135;
229 4          IL(1)=124;
230 4          IL(2)=071;
231 4          IL(3)=105;
232 4          IL(4)=127;
233 4          END;
234 3          DO;
235 4          OUTPUT(PORTB$RAM2)=1FH;
236 4          IG(0)=230;
237 4          IG(1)=230;
238 4          IG(2)=045;
239 4          IL(0)=135;
240 4          IL(1)=124;
241 4          IL(2)=071;
242 4          IL(3)=105;
243 4          IL(4)=127;
244 4          END;
245 3          ;
246 3          END;
247 2          PRIN3:      IF B=0 THEN DO;
249 3          PRIN4:      CALL TRANSMISION$AL$PC;
250 3          IF B='#' THEN CALL PCALFABETA;
252 3          END;
253 2          W=0;
254 2          DO I=0 TO 2;
255 3          IF IG(I)>=229 THEN
256 3          DO;
257 4          MO4:          IF P(I)=0 THEN P(I)=128;
259 4          MO5:          FLG(I)=1;
260 4          W=1;
261 4          END;
262 3          ELSE DO;
263 4          FLG(I)=0;
264 4          END;
265 3          END;
266 2          MO2:          IF W=1 THEN CALL RUTTESTEO(.P);

```

```
268 2      MO3:   DO I=0 TO 2;
269 3          IF IG(I)>= 195 THEN
270 3          DO;
271 4              P(I)=128 +(IG(I)-195)/3; /*Aproximaciones aceptables*/
272 4          END;
273 3          ELSE DO;
274 4              P(I)=0;
275 4          END;
276 3      MO10:  END;
277 2          CALL PRU1;
278 2          GOTO PRINCIPIO;
279 2      MO11:  END;
280 1      END ESTABILIDAD;
```

MODULE INFORMATION:

```
CODE AREA SIZE      = 0448H    1096D
VARIABLE AREA SIZE = 001AH     26D
MAXIMUM STACK SIZE = 0004H     4D
291 LINES READ
0 PROGRAM ERROR(S)
```

END OF PL/M-80 COMPILATION

ISIS-II PL/M-80 V3.1 COMPILATION OF MODULE MODTESTEO
 OBJECT MODULE PLACED IN RUTTEW.OBJ
 COMPILER INVOKED BY: PLM80 RUTTEW.PLM WORKFILES(:F0:,:F0:)
 -DEBUG PAGELength(54) PAGEWIDTH(60)

```

          /* Rutina de testeo durante un tiempo
-   T que depende
          del valor de P(I) */
1      MOD$TESTEO:DO;
2      1      DECLARE T(3) ADDRESS PUBLIC;
3      1      DECLARE IG(3) BYTE EXTERNAL;
4      1      DECLARE IL(5) BYTE EXTERNAL;
5      1      DECLARE FLG(3) BYTE EXTERNAL;
6      1      DECLARE (ALFA,BETA,INOMINAL,SI) BYTE EXTER
-   NAL;

7      1      DESCARGA:PROCEDURE EXTERNAL;
8      2      END DESCARGA;

9      1      TRANSMISION$AL$PC:PROCEDURE EXTERNAL;
10     2      END TRANSMISION$AL$PC;

11     1      RUTTESTEO:PROCEDURE (DIRPI) PUBLIC;
12     2      DECLARE (DIRPI) ADDRESS;
13     2      DECLARE (A,K,F,J,CONTADOR) BYTE;
14     2      DECLARE IN LITERALLY '212';
15     2      DECLARE PORTA$RAM1 LITERALLY '41H';
16     2      DECLARE P BASED DIRPI (3) BYTE;
17     2      TE1: DO;
18     3      CONTADOR=0; /*Contador para transmissio
-   n al PC */
19     3      DO J=0 TO 2;
20     4      IF FLG(J)=1 THEN DO;
22     5      IF P(J)>138 THEN P(J)=138;
24     5      T(J)=1471-(1470*(P(J)-128))/10; /*Se c
-   alcula un T al que este mal.Este T se ira dec
-   rementando*/
25     5      TE3: END;
26     4      ELSE T(J)=1500; /*Se inicializan l
-   os otros T's */
27     4      END;
28     3      A=00H;
29     3      BASE: DO J=0 TO 2;
30     4      IF FLG(J)=0 THEN
31     4      DO;
32     5      IF IG(J)>=229 THEN
33     5      DO;
34     6      IF P(J)=0 THEN P(J)=128;
36     6      P(J)=P(J)+(IG(J)-195)/3;

```

```

37 6          IF P(J)>138 THEN P(J)=138;
39 6          T(J)=1471-(1470*(P(J)-128))/10;
40 6          FLG(J)=1;
41 6          END;
42 5          ELSE DO;
43 6          IF IG(J)>=195 THEN P(J)=128+(I
- G(J)-195)/3;
45 6          ELSE P(J)=0;
46 6          END;
47 5          END;
48 4          END;
49 3          F=0;
50 3          DO J=0 TO 2;
51 4              IF FLG(J)=1 THEN DO;
53 5                  IF T(J)=0 AND IG(J)>=229 THEN F=1;
55 5              END;
56 4          END;
57 3          TE2:  IF F=1 THEN DO;
59 4              DO J=0 TO 2;
60 5                  FLG(J)=0;
61 5              END;
62 4              CALL DESCARGA;
63 4              GOTO TE10;
64 4          END;
65 3          A=0;
66 3          DO J=0 TO 2;
67 4              IF FLG(J)=1 THEN DO;
69 5                  IF T(J)=0 AND IG(J)<229 THEN FLG(J)=0;
71 5                  ELSE A=1;
72 5              END;
73 4          END;

/*Realizar una transmision al PC cada 20 v
- eces que se repita el bucle*/
74 3          IF CONTADOR=20 THEN DO;
76 4              CONTADOR=0;
77 4              K=INPUT(PORTA$RAM1) AN
- D 01H;
78 4              IF K=0 THEN CALL TRANSMISION$A
- L$PC;
80 4          IF SI=1 THEN DO;
82 5              IG(0)=209;
83 5              IG(1)=189;
84 5              IG(2)=189;
85 5              IL(0)=148;
86 5              IL(1)=105;
87 5              IL(2)=145;
88 5              IL(3)=125;
89 5              IL(4)=115;
90 5              SI=8;
91 5          END;
92 4          END;

```

```
93 3          CONTADOR=CONTADOR+1;
94 3          IF A=1 THEN GOTO BASE;
96 3      TE10:  RETURN;
97 3          END;
98 2          END RUTTESTEO;
99 1          END MOD$TESTEO;
```

MODULE INFORMATION:

```
CODE AREA SIZE      = 02E9H      745D
VARIABLE AREA SIZE = 000DH       13D
MAXIMUM STACK SIZE = 0006H       6D
91 LINES READ
0 PROGRAM ERROR(S)
```

END OF PL/M-80 COMPILATION

ISIS-II PL/M-80 V3.1' COMPILATION OF MODULE MODDESCARGA
 OBJECT MODULE PLACED IN DESCARW.OBJ
 COMPILER INVOKED BY: PLM80 DESCARW.PLM WORKFILES(:F0:,:F0:)
 (-DEBUG PAGELENGTH(54) PAGEWIDTH(60))

```

      /* Esta rutina tiene por objeto realizar :
-   a descarga de lineas
      con el siguiente criterio: Calculo la linea
-   a o conjunto de lineas
      mas cercana al 10% de la potencia nominal
-   de los generadores*/
1      MOD$DESCARGA:DO;
2      1      TRANSMISION$AL$PC:PROCEDURE EXTERNAL;
3      2      END TRANSMISION$AL$PC;

4      1      DECLARE IG(3) BYTE EXTERNAL;
5      1      DECLARE IL(5) BYTE EXTERNAL;
6      1      DECLARE (LINDESCONECTADAS,ALFA,BETA,INOMIN
-   AL,COR,SI) BYTE EXTERNAL;

7      1      DESCARGA:PROCEDURE PUBLIC;
8      2      DECLARE (ALFA,BETA,INOMIN,INOMIN,INOMIN)
9      2      DECLARE K BYTE;
10     2      DECLARE R(5) BYTE;
11     2      DECLARE (I,J,X,DESCONECTO,INTERMEDIA,
-   INTER,DES,V1) BYTE;
12     2      DECLARE PORTA$RAM1 LITERALLY '41H';
13     2      DECLARE PORTB$RAM1 LITERALLY '42H';
14     2      DECLARE PORTB$RAM2 LITERALLY '32H';
15     2      DECLARE CTE LITERALLY '45';
16     2      DECLARE IN$LINEA LITERALLY '153';
17     2      DECLARE GUARDALINEAS(5) BYTE;

18     2      V1=0;

      /* Ordeno las lineas de menor a mayor y qu
-   ando en R el numero de
      de cada linea */
19     2      DECA: DO;
20     3      DO I=0 TO 4; /*Inicializo R(I) con
-   su valor de origen*/
21     4      R(I)=IL(I);
22     4      GUARDALINEAS(I)=I;
23     4      END;
24     3      DE1: DO I=0 TO 3; /*Ordenamiento*/
25     4      DO X=I+1 TO 4;
26     5      IF R(X)<R(I) THEN
27     5      DO;
28     6      INTER=GUARDALINEAS(X);

```

```

29 6          INTERMEDIA=R(X);
30 6          GUARDALINEAS(X)=GUARDALINEAS(I);
31 6          R(X)=R(I);
32 6          GUARDALINEAS(I)=INTER;
33 6          R(I)=INTERMEDIA;
34 6          END;
35 5          END;
36 4          END;
          /*Con los valores de ALFA y BETA recibidos e
-         stablezco una forma de hallar
-         el 10% de la corriente nominal de los gene
-         radores*/

37 3          DE2:      D=(10+ALFA+BETA)*IN$LINEA;

          /* Establezco una forma de eleccion de las l
-         inas a desconectar*/

38 3          DE3:      DO I=0 TO 4;
39 4              IF R(I)<45 THEN R(I)=45;
41 4              END;
42 3          DE4:      X=0;
43 3              CALCULO=(R(X)-CTE)*100;
44 3              DO WHILE CALCULO<D;
45 4                  X=X+1;
46 4                  CALCULO=(R(X)-CTE)*100;
47 4              END;

48 3          DE5:      IF X>4 THEN X=4;

50 3          DE6:      DO CASE GUARDALINEAS(X);          /*GUARDALI
-         NEAS(X) contiene un numero comprendido entre
-         0 y 4*/
51 4              DESCONECTO=11101111B;
52 4              DESCONECTO=11110111B;
53 4              DESCONECTO=11111011B;
54 4              DESCONECTO=11111101B;
55 4              DESCONECTO=11111110B;
56 4              END;
57 3              IF IL(GUARDALINEAS(X))<=45 THEN DO;
59 4                  OUTPUT(PORTB$RAM1)=03H;
60 4                  COR=01H;
61 4                  END;
62 3          DE7:      OUTPUT(PORTB$RAM2)=LINDESCONECTA
-         DAS AND DESCONECTO;
63 3              IF SI=2 THEN DO;
65 4                  IG(0)=209;
66 4                  IG(1)=189;
67 4                  IG(2)=189;
68 4                  IL(0)=148;
69 4                  IL(1)=45;

```

```
70 4 IL(2)=205;
71 4 IL(3)=125;
72 4 IL(4)=115;
73 4 SI=8;
74 4 END;
75 3 IF SI=3 THEN DO;
77 4 IG(0)=189;
78 4 IG(1)=209;
79 4 IG(2)=189;
80 4 IL(0)=148;
81 4 IL(1)=45;
82 4 IL(2)=205;
83 4 IL(3)=125;
84 4 IL(4)=115;
85 4 SI=8;
86 4 END;
87 3 IF SI=4 THEN DO;
89 4 IG(0)=189;
90 4 IG(1)=189;
91 4 IG(2)=209;
92 4 IL(0)=148;
93 4 IL(1)=45;
94 4 IL(2)=205;
95 4 IL(3)=125;
96 4 IL(4)=115;
97 4 SI=8;
98 4 END;
99 3 IF SI=5 THEN DO;
101 4 IG(0)=045;
102 4 IG(1)=196;
103 4 IG(2)=196;
104 4 IL(0)=135;
105 4 IL(1)=124;
106 4 IL(2)=071;
107 4 IL(3)=045;
108 4 IL(4)=127;
109 4 SI=8;
110 4 END;
111 3 IF SI=6 THEN DO;
113 4 IG(0)=196;
114 4 IG(1)=045;
115 4 IG(2)=196;
116 4 IL(0)=135;
117 4 IL(1)=124;
118 4 IL(2)=071;
119 4 IL(3)=045;
120 4 IL(4)=127;
121 4 SI=8;
122 4 END;
123 3 IF SI=7 THEN DO;
125 4 IG(0)=196;
```



```

126 4          IG(1)=196;
127 4          IG(2)=045;
128 4          IL(0)=135;
129 4          IL(1)=124;
130 4          IL(2)=071;
131 4          IL(3)=045;
132 4          IL(4)=127;
133 4          SI=8;
134 4          END;
135 3          DO I=1 TO 4;
136 4              CALL TIME(250);
137 4          END;
138 3          IF COR=01H THEN GOTO DE11;
140 3          DEB:          CALL TRANSMISION$AL$PC;
141 3              DO I=1 TO 14;          /*Retardo de 350 m
-          seg + tiempo de */
142 4              CALL TIME(250);          /*ejecucion de pro
-          grama ~ 0.5 seg */
143 4          END;

144 3          DE9:          DES=00;
145 3              DO I=0 TO 2;
146 4                  IF IG(I)>=229 THEN DES=01; /*Despues d
-          e 0.5 seg de desconectada la linea */
148 4                  END;          /*no se han
-          recuperado los generadores*/
149 3                  K=INPUT(PORTA$RAM1) AND 01H;
150 3          DE10:          IF DES=1 THEN
151 3          DE12:          DO;
152 4          DE13:          IF K=0 THEN CALL TRANSMISION$AL$PC
-          ;
154 4          DE14:          GOTO DECA;
155 4          DE15:          END;
156 3          DE11:          IF K=0 THEN CALL TRANSMISION$AL$PC
-          ;
158 3          DE16:          RETURN;
159 3          END;

160 2          END DESCARGA;
161 1          END MOD$DESCARGA;

```

MODULE INFORMATION:

```

CODE AREA SIZE      = 036EH      678D
VARIABLE AREA SIZE = 0017H      23D
MAXIMUM STACK SIZE = 0002H      2D
168 LINES READ
0 PROGRAM ERROR(S)
PL/M-80 COMPILER

```

ISIS-II PL/M-80 V3.1 COMPILATION OF MODULE TOMADEDATOS
 OBJECT MODULE PLACED IN TDATAW.OBJ
 COMPILER INVOKED BY: PLM80 TDATAW.PLM WORKFILES(:F0:,;F0:)
 -DEBUG PAGELLENGTH(54) PAGEWIDTH(60)

```

1          TOMA$DE$DATOS:DO;
2      1          DECLARE IG(3) BYTE EXTERNAL;
3      1          DECLARE IL(5) BYTE EXTERNAL;
4      1
5      1          DECLARE (INIC,LINDESCONECTADAS,CANALDA
-      C) BYTE EXTERNAL;
6      1          DECLARE FLG(3) BYTE EXTERNAL;
7      1          DECLARE T(3) ADDRESS EXTERNAL;
8      1          TDATA$PROCEDURE PUBLIC;
9      2          DECLARE PORTA$RAM2 LITERALLY '31H';
10     2          DECLARE PORTB$RAM2 LITERALLY '32H';
11     2          DECLARE (B,I,PEPE) BYTE;
12     2          DECLARE ADC(8) BYTE AT (2000H);
13     2          T1:      DO;
14     3              DO I=0 TO 2;
15     4                  IF FLG(I)=1 THEN DO;
17     5                      T(I)=T(I)-1;
18     5
20     5                  END;
21     4          END;
22     3          T2:      IF CANALDAC<3 THEN DO;
24     4                  PEPE=ADC(CANALDAC);
25     4                  END;
26     3                  ELSE DO;
27     4                      PEPE=ADC(CANALDAC);
28     4                  END;
29     3
30     3          T3:      IF CANALDAC=0BH THEN DO;
32     4                  LINDESCONECTADAS=INPUT(PORTA$RAM2)
-      ;
33     4                  CANALDAC=0;
34     4                  INIC=0;
35     4                  END;
36     3                  ADC(CANALDAC)=0;
37     3          T10:     RETURN;
38     3          END;
39     2          END TDATA$;
40     1          END TOMA$DE$DATOS;

```

MODULE INFORMATION:
PL/M-80 COMPILER

PAGE 2

CODE AREA SIZE = 00B7H 183D
VARIABLE AREA SIZE = 0003H 3D
MAXIMUM STACK SIZE = 0002H 2D

! 37 LINES READ
! 0 PROGRAM ERROR(S)

END OF PL/M-80 COMPILATION

LOC	OBJ		LINE	SOURCE STATEMENT
			1	EXTRN TDATOS, COMIEN
0000			2	ORG 0
0000	C50000	E	3	JMP COMIEN
0034			4	ORG 34H
0034	C50000	C	5	JMP RI1
			6	CSEG
0000	F3		7	RI1: DI
0001	F5		8	PUSH PSW
0002	E5		9	PUSH H
0003	D5		10	PUSH D
0004	C5		11	PUSH E
0005	CD0000	E	12	CALL TDATOS
0008	C1		13	POP B
0009	D1		14	POP D
000A	E1		15	POP H
000B	F1		16	POP PSW
000C	FB		17	EI
000D	C9		18	RI10: RET
			19	END

PUBLIC SYMBOLS

EXTERNAL SYMBOLES

COMIEN E 0000 TDATOS E 0000

USER SYMBOLS

COMIEN E 0000 RI1 C 0000 RI10 C 000D TDATOS E 0000

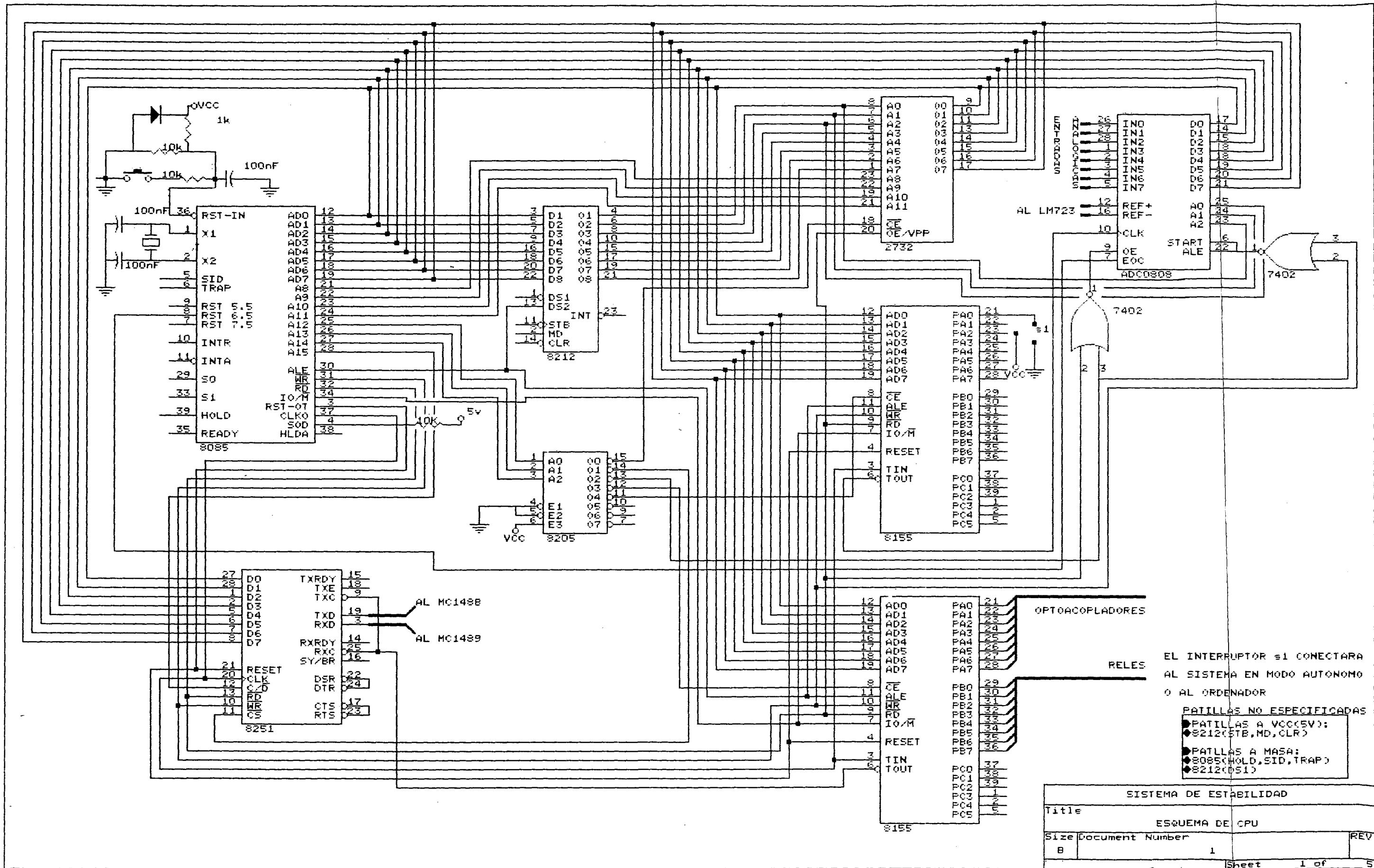
ASSEMBLY COMPLETE, NO ERRORS

LOC	OBJ	LINE	SOURCE STATEMENT
		1	PUBLIC PRU1
		2	EXTRN CI,CST5,SI
		3	CSEG
FC9F		4	CO EQU 0FC9FH
0000	FD	5	PRU1: PUSH PSW
0001	ED	6	PUSH H
0002	DD	7	PUSH D
0003	CD	8	PUSH B
0004	0E1D	9	MVI C,01DH
0006	CD9FFC	10	CALL CO
0009	0E00	11	MVI C,0
000B	CD9FFC	12	CALL CO
000E	0E1B	13	MVI C,01BH
0010	CD9FFC	14	CALL CO
0013	0E45	15	MVI C,045H
0015	CD9FFC	16	CALL CO
0018	213D00	17	C LXI H,T1
001E	4E	18	TX: MOV C,M
001C	CD9FFC	19	CALL CO
001F	23	20	INX H
0020	79	21	MOV A,C
0021	FE2E	22	CPI '.'
0023	C21B00	23	C JNZ TX
0026	CD0000	24	E T2: CALL CST5
0029	D6FF	25	SUI 0FFH
002B	C23B00	26	C JNZ T3
002E	CD0000	27	E CALL CI
0031	320000	28	E STA SI
0034	4F	29	MOV C,A
0035	CD9FFC	30	CALL CO
0038	C1	31	T3: POP B
0039	E1	32	POP H
003A	D1	33	POP D
003B	C1	34	POP B
003C	C9	35	RET
003D	45534352	36	T1: DB 'ESCRIBE LA NUEVA OPCION.'
0047	53455641		
004D	204F5043		
0051	494F4E2E		
		37	END

LOC	OBJ	LINE	SOURCE STATEMENT								
PUBLIC SYMBOLS											
FRU1	C	0000									
EXTERNAL SYMBOLS											
CI	E	0000	CSTS	E	0000	SI	E	0000			
USER SYMBOLS											
CI	E	0000	CO	A	FC9F	CSTS	E	0000	FRU1	C	0000
SI	E	0000	T1	C	003D	T2	C	0026	T3	C	0038
TX	C	001E									
ASSEMBLY COMPLETE, NO ERRORS											

A P E N D I C E

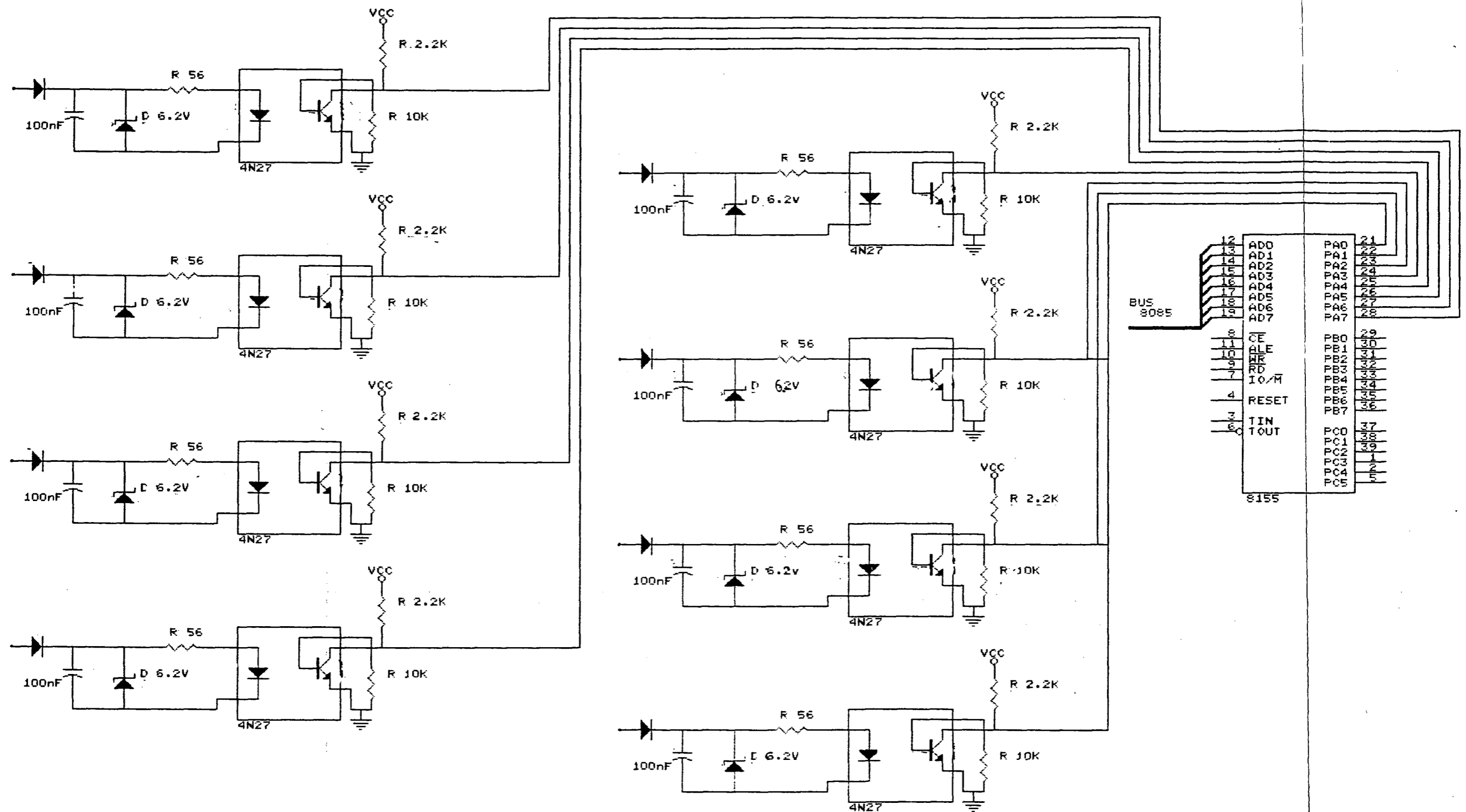
P L A N O S



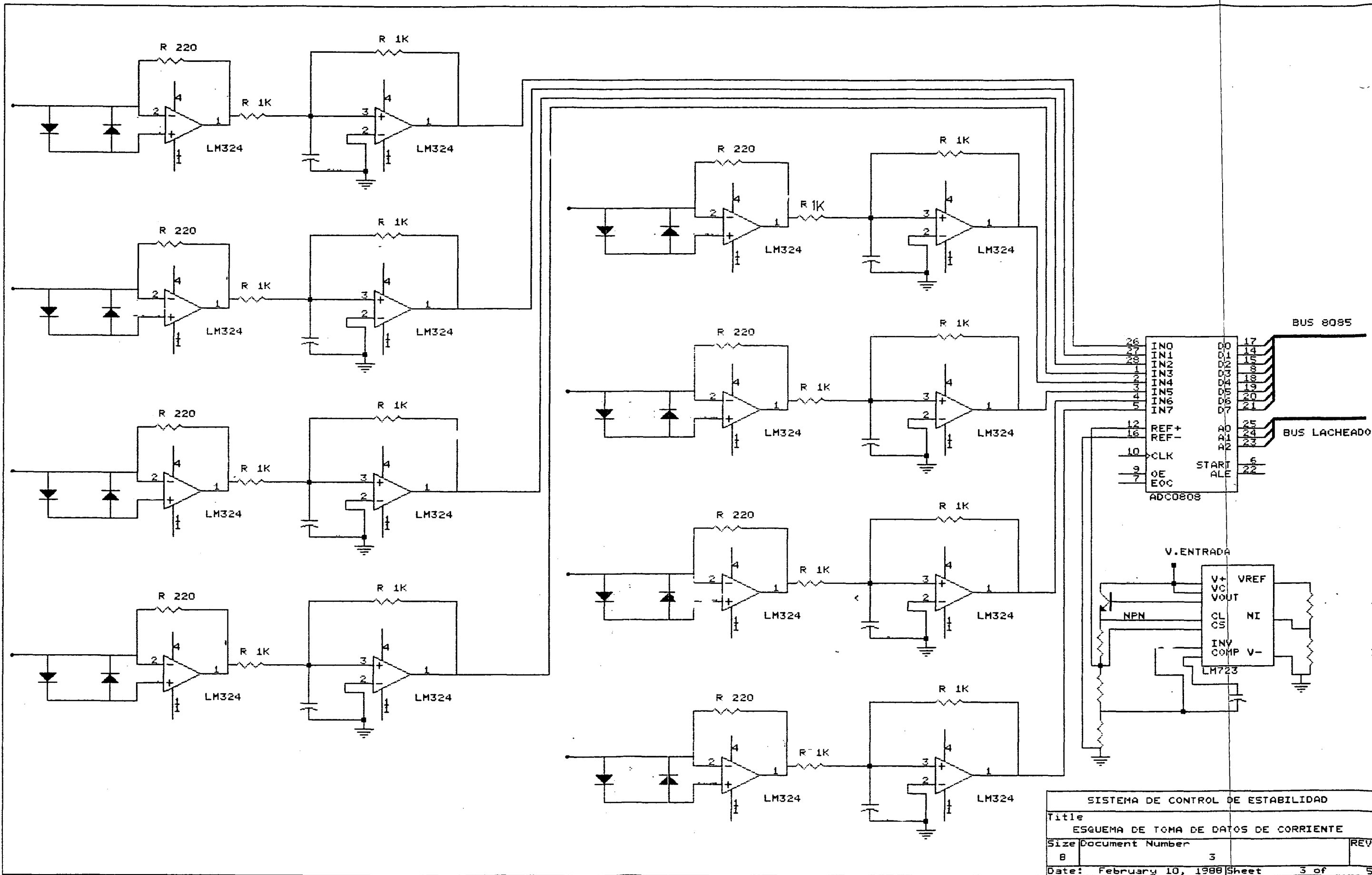
EL INTERRUPTOR s1 CONECTARA AL SISTEMA EN MODO AUTONOMO O AL ORDENADOR

- PATILLAS NO ESPECIFICADAS
- PATILLAS A VCC(5V):
 - 8212(5TB, MD, CLR)
 - PATILLAS A MASA:
 - 8085(HOLD, SID, TRAP)
 - 8212(0S1)

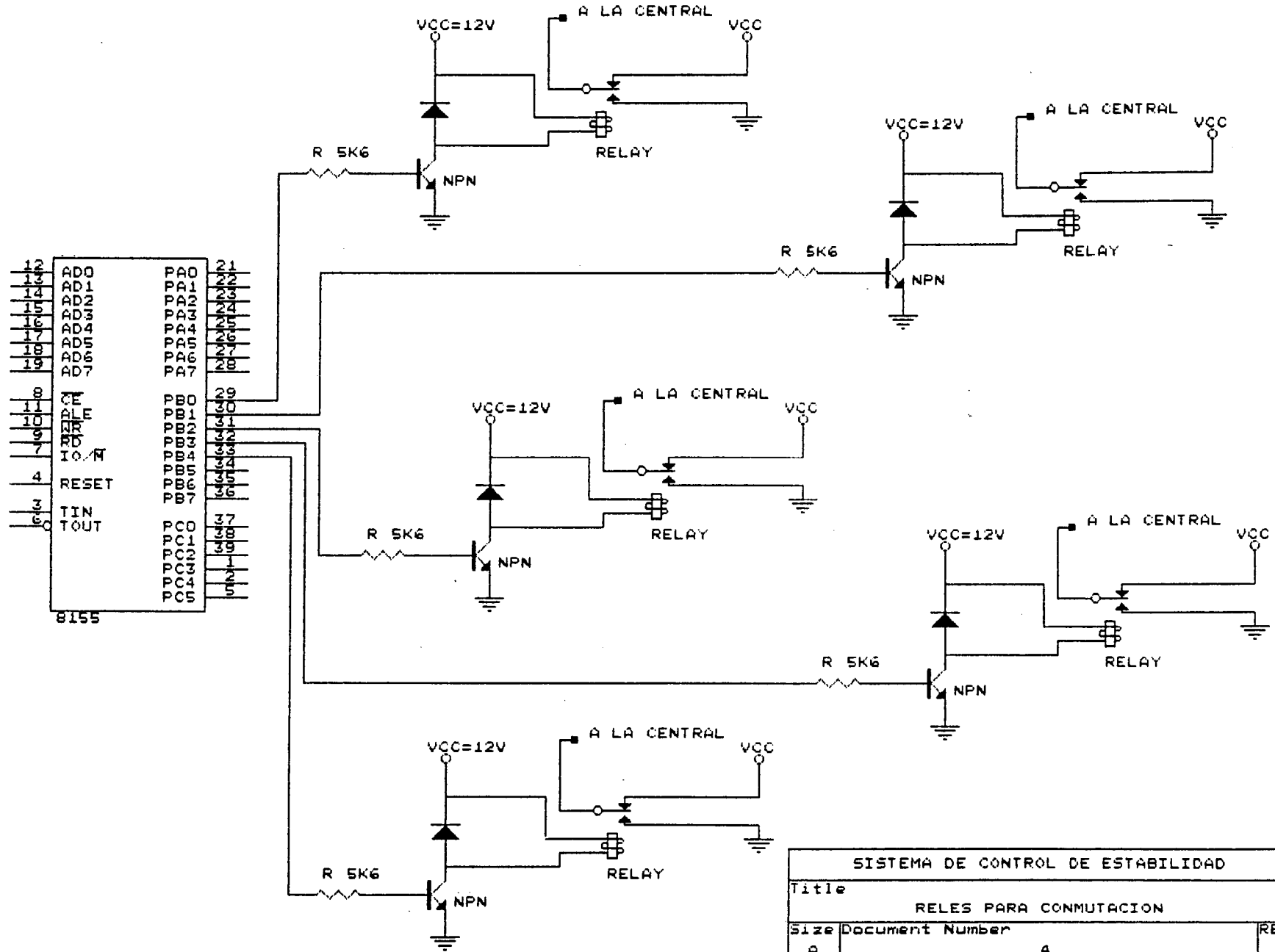
SISTEMA DE ESTABILIDAD			
Title			
ESQUEMA DE CPU			
Size	Document Number	REV	
B	1		
Sheet			1 of 5



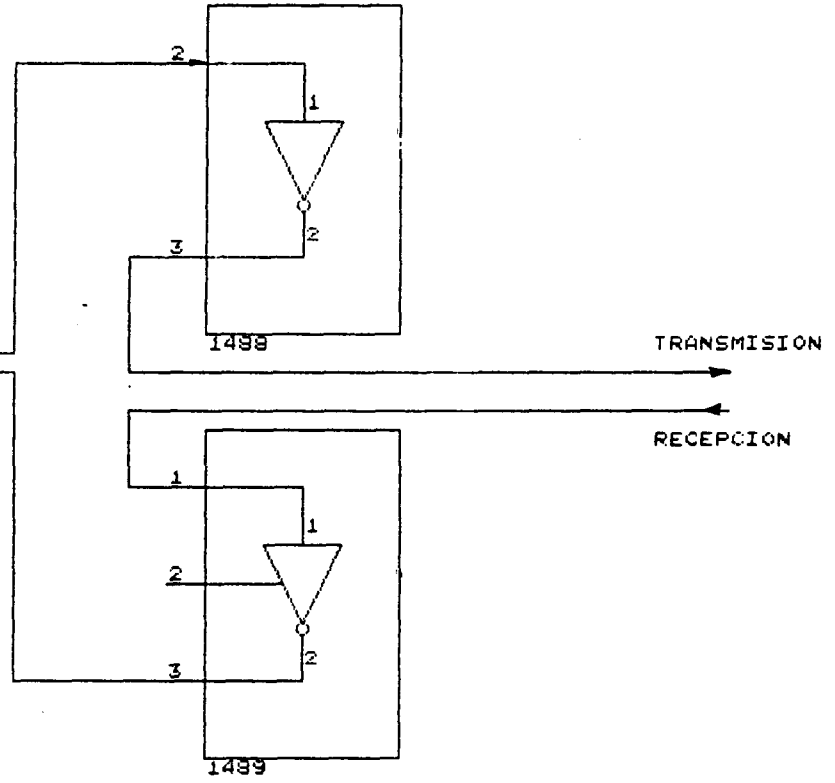
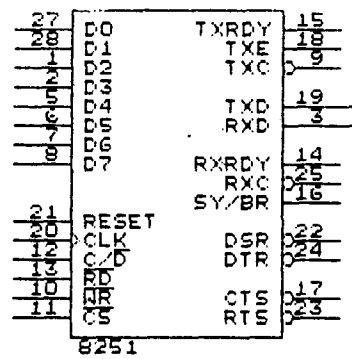
SISTEMA DE CONTROL DE ESTABILIDAD		
Title		
TOMA DE DATOS SOBRE ESTADOS		
Size	Document Number	REV
B	2	
Sheet		2 of 5



SISTEMA DE CONTROL DE ESTABILIDAD		
Title ESQUEMA DE TOMA DE DATOS DE CORRIENTE		
Size B	Document Number 3	REV
Date: February 10, 1988	Sheet	3 of 5



SISTEMA DE CONTROL DE ESTABILIDAD		
Title		
RELES PARA CONMUTACION		
Size	Document Number	REV
A	4	
Sheet		4 of 5



SISTEMA DE CONTROL DE ESTABILIDAD		
Title		
TRANSMISION DE DATOS		
Size	Document Number	REV
A	5	
Sheet		of 5

C A R A C T E R I S T I C A S D E
C O M P O N E N T E S

RAD-HARD CMOS COMMUNICATION PRODUCTS

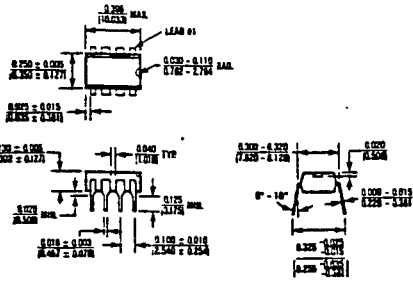
Manchester Encoder/Decoder (MED) HS-15530RH		
Features	Radiation Effects	Pinout
<ul style="list-style-type: none"> Specifically designed for radiation hardness Support of MIL-STD-1553 1.0 Mbit/sec data rate Sync identification and lock-in Clock recovery Manchester II encode, decode Separate encode and decode Low operating power: 50 mW @ 5 V Full military temperature range 	<ul style="list-style-type: none"> Each lot screened for total dose hardness Parametrics guaranteed to 1×10^5 rads (Si) Latch-up free Upset; $> 10^8$ rads (Si)/sec 	

HS-245 Triple Line Transmitter, HS-246/249 Triple Line Receivers, HS-248 Triple Party-Line Receiver		
Features		
<ul style="list-style-type: none"> High speed: 15 MHz with 50-ft. cable, 2 MHz with 1,000-ft. cable Tolerates -2.0 V to $+20.0$ V ground differential (transmitter with respect to receiver) Current mode operation High common mode rejection Transmitter and receiver party-line capability Transmitter input/receiver output TTL/DTL compatible Low power dissipation Low EMI generation High noise immunity Replaces HD-245/246/248/249 		
Pinouts		
<p> 1 INPUT 1 2 OUTPUT 2 3 OUTPUT 3 4 INPUT 4 5 INPUT 5 6 OUTPUT 6 7 SUBSTRATE GND 8 INPUT 8 9 OUTPUT 9 10 INPUT 10 11 OUTPUT 11 12 OUTPUT 12 13 INPUT 13 14 V_{CC} </p>	<p> 1 (-) INPUT 1 2 (+) INPUT 2 3 (R1) OUTPUT 3 4 (-) INPUT 4 5 (+) INPUT 5 6 (R2) OUTPUT 6 7 GND 8 INPUT (-) 9 INPUT (+) 10 OUTPUT R3 11 V_{CC} R3 12 V_{CC} R1 & R2 13 V_{CC} R3 14 V_{CC} R1 & R2 </p>	
<p>HS-245 Transmitter HS-246/248/249 Receivers</p>		

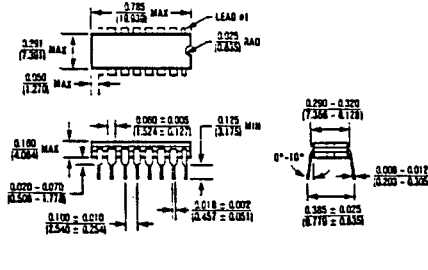
SECURE DATA COMMUNICATIONS

CYPHER-I™ CMOS DATA ENCRYPTION DEVICE HS-3447		Pinout
Features		
<ul style="list-style-type: none"> Endorsed by National Security Agency for protecting unclassified national security related information (UNSR) Per DoD drawing ON304455 Alternative to WD 2001/2002 and MC6859 NMOS devices Uses single 5V power supply Operating range -55°C to $+125^{\circ}\text{C}$ Lower power operation 250 mW at 10 MHz Maximum transfer rate: <ul style="list-style-type: none"> 20 MHz at 7 Volts 10 MHz at 5 Volts 20 MHz at 5 Volts (-55°C to $+85^{\circ}\text{C}$) Encrypts/decrypts via serial data stream Available to Class B and Class S equiv. screening Inputs TTL compatible Key variable stored on chip is not externally accessible Available in special configurations 		

Microprocessor Supervisory Circuits



8 Lead Plastic DIP (PA)
 $\theta_{JA} = 120^{\circ}\text{C/W}$
 $\theta_{JC} = 70^{\circ}\text{C/W}$

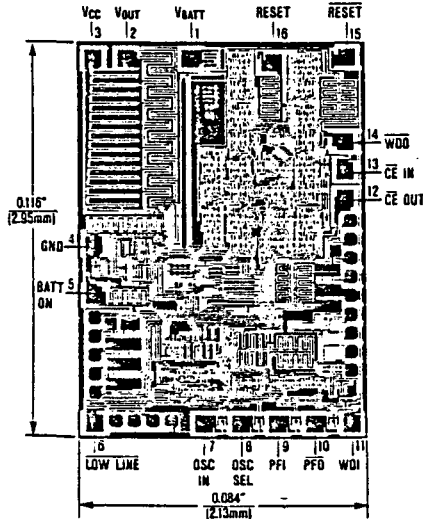


16 Lead CERDIP (JE)
 $\theta_{JA} = 100^{\circ}\text{C/W}$
 $\theta_{JC} = 50^{\circ}\text{C/W}$

Ordering Information

PART	TEMP. RANGE	PACKAGE
MAX692CPA	0°C to +70°C	8 Lead Plastic DIP
MAX692EPA	-40°C to +85°C	8 Lead Plastic DIP
MAX692EJA	-40°C to +85°C	8 Lead CERDIP
MAX692MJA	-55°C to +125°C	8 Lead CERDIP
MAX693C/D	0°C to +70°C	Dice
MAX693CPE	0°C to +70°C	16 Lead Plastic DIP
MAX693CWE	0°C to +70°C	16 Lead Small Outline
MAX693EPE	-40°C to +85°C	16 Lead Plastic DIP
MAX693EJE	-40°C to +85°C	16 Lead CERDIP
MAX693EWE	-40°C to +85°C	16 Lead Small Outline
MAX693MJE	-55°C to +125°C	16 Lead CERDIP

Chip Topography



Maxim cannot assume responsibility for use of any circuitry other than circuitry entirely enclosed in a Maxim product. No circuit patent licenses are implied. Maxim reserves the right to change the circuitry and specifications without notice at any time.

16 Maxim Integrated Products, 510 N. Pastora Avenue, Sunnyvale, CA 94086 (408) 737-7600

MAXIM INTEGRATED PRODUCTS

MAXIM Microprocessor Supervisory Circuits

Features

- ◆ Precision Voltage Monitor
4.65V in MAX690 and MAX691
4.40V in MAX692 and MAX693
- ◆ Power OK/Reset Time Delay
- ◆ Watchdog Timer—100ms, 1.6sec, or adjustable
- ◆ Minimum Component Count
- ◆ 1µA Standby Current
- ◆ Battery Backup Power Switching
- ◆ Onboard Gating of Chip Enable Signals
- ◆ Voltage Monitor for Power Fail or Low Battery Warning

General Description

The MAX690 Family of supervisory circuits reduce the complexity and number of components required for power supply monitoring and battery control functions in microprocessor systems. These include µP reset and backup-battery switchover, watchdog timer, CMOS RAM write protection, and power-failure warning. The MAX690 Family significantly improves system reliability and accuracy compared to that obtainable with separate ICs or discrete components.

The MAX690 and MAX692 are supplied in 8-pin packages and provide four functions:

- 1) A Reset output during power-up, power-down and brownout conditions.
- 2) Battery backup switching for CMOS RAM, CMOS microprocessor or other low power logic.
- 3) A Reset pulse if the optional watchdog timer has not been toggled within a specified time.
- 4) A 1.25V threshold detector for power fail warning, low battery detection, or to monitor a power supply other than +5V.

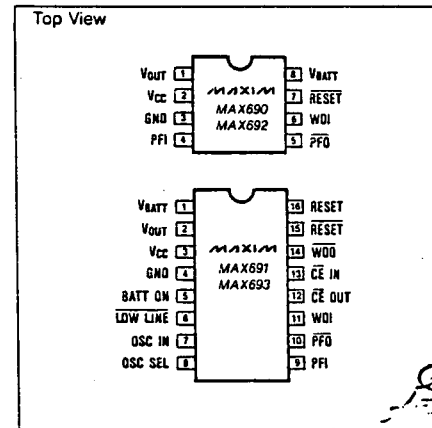
The MAX691 and MAX693 are supplied in 16-pin packages and perform all MAX690/692 functions, plus:

- 1) Write protection of CMOS RAM or EEPROM.
- 2) Adjustable reset and watchdog timeout periods.
- 3) Separate outputs for indicating a watchdog timeout, backup battery switchover, and low V_{CC}.

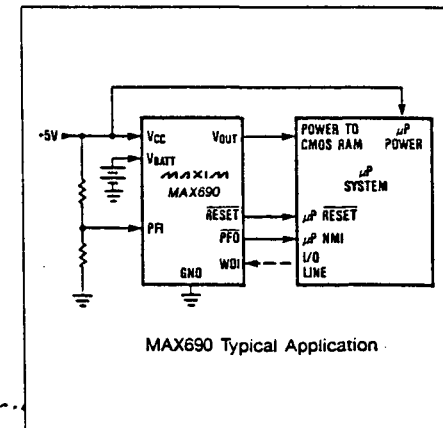
Applications

- Computers
- Controllers
- Intelligent Instruments
- Automotive Systems
- Critical µP Power Monitoring

Pin Configurations



Typical Operating Circuit



MAX690 Typical Application

AVDA. DE VALLADOLID, 27
 28008 MADRID TX, 47397
 TEL: 349 33 55 342 55 66

MAXIM
 MAXIM is a registered trademark of Maxim Integrated Products.

Maxim Integrated Products 1

Microprocessor Supervisory Circuits

ABSOLUTE MAXIMUM RATINGS

Terminal Voltage (with respect to GND)	-0.3V to 6.0V
V _{CC}	-0.3V to 6.0V
V _{BATT}	-0.3V to 6.0V
All Other Inputs (Note 1)	-0.3V to (V _{OUT} +0.5V)
Input Current	
V _{CC}	200mA
V _{BATT}	50mA
GND	20mA
Output Current	
V _{OUT}	short circuit protected
All Other Outputs	20mA
Rate-of-Rise, V _{BATT} , V _{CC}	100V/μs
Operating Temperature Range	
C suffix	0°C to +70°C
E suffix	-40°C to +85°C
M suffix	-55°C to +125°C

Power Dissipation	
8 Pin Plastic DIP	
(Derate 5mW/°C above +70°C)	400mW
8 Pin CERDIP	
(Derate 8mW/°C above +85°C)	500mW
16 Pin Plastic DIP	
(Derate 7mW/°C above +70°C)	600mW
16 Pin Small Outline	
(Derate 7mW/°C above +70°C)	600mW
16 Pin CERDIP	
(Derate 10mW/°C above +85°C)	600mW
Storage Temperature Range	-65°C to +160°C
Lead Temperature (Soldering, 10 seconds)	300°C

ELECTRICAL CHARACTERISTICS

(V_{CC} = full operating range, V_{BATT} = 2.8V, T_A = 25°C, unless otherwise noted.)

PARAMETER	CONDITIONS	MIN	TYP	MAX	UNITS
BATTERY BACKUP SWITCHING					
Operating Voltage Range		4.75		5.5	V
MAX690, MAX691 V _{CC}		2.0		4.25	
MAX690, MAX691 V _{BATT}		4.5		5.5	
MAX692, MAX693 V _{CC}		2.0		4.0	
MAX692, MAX693 V _{BATT}					
V _{OUT} Output Voltage	I _{OUT} = 1mA I _{OUT} = 50mA		V _{CC} -0.1 V _{CC} -0.25		V
V _{OUT} in Battery Backup Mode	I _{OUT} = 100μA, V _{CC} < V _{BATT} - 0.2V		V _{BATT} - 0.1, V _{BATT} - 0.02		V
Supply Current (excludes I _{OUT})	I _{OUT} = 1mA I _{OUT} = 100mA		4 10		mA
Supply Current in Battery Backup Mode	V _{CC} = 0V, V _{BATT} = 2.8V		0.6	1	μA
Battery Standby Charging Current	5.5V > V _{CC} > V _{BATT} + 0.2V I _{OUT} = 1mA I _{OUT} = 100mA			1 5	μA
Battery Switchover Threshold	Power Up Power Down		70 50		mV
Battery Switchover Hysteresis			20		mV
BATT ON Output Voltage	I _{SINK} = 3.2mA			0.4	V
BATT ON Output Short Circuit Current	BATT ON = V _{OUT} BATT ON = 0V	0.5	7 1	25	mA μA
RESET AND WATCHDOG TIMER					
Reset Voltage Threshold	MAX690, MAX691 MAX692, MAX693	4.5 4.25	4.65 4.4	4.75 4.5	V

Note 1. The input voltage limits on PFI and WDI may be exceeded provided the input current is limited to less than 10mA.

Microprocessor Supervisory Circuits

ELECTRICAL CHARACTERISTICS (continued)

(V_{CC} = full operating range, V_{BATT} = 2.8V, T_A = 25°C, unless otherwise noted.)

PARAMETER	CONDITIONS	MIN	TYP	MAX	UNITS
Reset Threshold Hysteresis			40		mV
Reset Timeout Delay	Figure 6, OSC SEL HIGH	35	50	70	ms
Watchdog Timeout Period, Internal Oscillator	Long Period Short Period	1.0 70	1.6 100	2.25 140	sec ms
Watchdog Timeout Period, External Clock	Long Period Short Period	4032 960		4097 1025	Clock Cycles
Minimum WDI Input Pulse Width	V _{IL} = 0.4, V _{IH} = 3.5V	200			ns
RESET and LOW LINE Output Voltage	I _{SINK} = 1.6mA I _{SOURCE} = 1μA, V _{CC} = 5V	3.5		0.4	V
RESET and WDO Output Voltage	I _{SINK} = 800μA I _{SOURCE} = 1μA, V _{CC} = 5V	3.5		0.4	V
Output Short Circuit Current	RESET, RESET, WDO, LOW LINE	1	3	25	μA
WDI Input Threshold	V _{CC} = 5V (Note 2)			0.8	V
WDI Input Logic High		3.0			
WDI Mid-Level Logic Voltage	V _{CC} = 5V (Note 2)	1.3	1.9	2.5	V
WDI Input current	WDI = V _{OUT} WDI = 0V		20	-15	μA
POWER FAIL DETECTOR					
PFI Input Threshold		1.15	1.25	1.35	V
PFI Input Current			±0.01	±10	nA
PFO Output Voltage	I _{SINK} = 3.2mA I _{SOURCE} = 1μA			0.4 3.5	V
PFO Short Circuit Source Current	PFI = 0V, PFO = 0V	1	3	25	μA
CHIP ENABLE GATING					
CE IN Thresholds	V _{IL} V _{IH}	0.8		3.0	V
CE IN Pullup Current			3		μA
CE OUT Output Voltage	I _{SINK} = 3.2mA I _{SOURCE} = 3.0mA			0.4	V
CE Propagation Delay			50		ns
OSCILLATOR					
OSC IN Input Current			±2		μA
OSC SEL Input Pullup Current			5		μA
OSC IN Frequency Range	OSC SEL = 0V	0		250	kHz
OSC IN Frequency with External Capacitor	OSC SEL = 0V C _{OSC} = 47pF		2		kHz

Note 1. The input voltage limits on PFI and WDI may be exceeded provided the input current is limited to less than 10mA.

Note 2. WDI is guaranteed to be in the mid-level (inactive) state if WDI is floating and V_{CC} is in the operating voltage range. WDI is internally biased to 38% of V_{CC} with an impedance of approximately 125 kilohms.

Microprocessor Supervisory Circuits

MAX690/91/92/93

Pin Description

NAME	PIN		FUNCTION
	MAX690/692	MAX691/693	
V _{CC}	2	3	The +5V input.
V _{BATT}	8	1	Backup battery input. Connect to Ground if a backup battery is not used.
V _{OUT}	1	2	The higher of V _{CC} or V _{BATT} is internally switched to V _{OUT} . Connect V _{OUT} to V _{CC} if V _{OUT} and V _{BATT} are not used.
GND	3	4	0V ground reference for all signals.
RESET	7	15	RESET goes low whenever V _{CC} falls below either the reset voltage threshold or the V _{BATT} input voltage. The reset threshold is typically 4.65V for the MAX690 and MAX691, and 4.4V for the MAX692 and MAX693. RESET remains low for 50 ms after V _{CC} returns to 5V. RESET also goes low for 50ms if the Watchdog Timer is enabled but not serviced within its timeout period. The RESET pulse width can be adjusted as shown in Table 1.
WDI	6	11	The watchdog input, WDI, is a three level input. If WDI remains either high or low for longer than the watchdog timeout period, RESET pulses low and WDO goes low. The Watchdog Timer is disabled when WDI is left floating or is driven to mid-supply. The timer resets with each transition at the Watchdog Timer input.
PFI	4	9	PFI is the non-inverting input to the Power Fail Comparator. When PFI is less than 1.25V, PFO goes low. Connect PFI to GND or V _{OUT} when not used. See Figure 1.
PFO	5	10	PFO is the output of the Power Fail Comparator. It goes low when PFI is less than 1.25V. The comparator is turned off and PFO goes low when V _{CC} is below V _{BATT} .
CE IN	-	13	The input to the CE gating circuit. Connect to GND or V _{OUT} if not used.
CE OUT	-	12	CE OUT goes low only when CE IN is low and V _{CC} is above the reset threshold (4.65V for MAX691, 4.4V for MAX693). See Figure 6.
BATT ON	-	5	BATT ON goes high when V _{OUT} is internally switched to the V _{BATT} input. It goes low when V _{OUT} is internally switched to V _{CC} . The output typically sinks 7mA and can directly drive the base of an external PNP transistor to increase the output current above the 100mA rating of V _{OUT} .
LOW LINE	-	6	LOW LINE goes low when V _{CC} falls below the reset threshold. It returns high as soon as V _{CC} rises above the reset threshold. See Figure 6, Reset Timing.
RESET	-	16	RESET is an active high output. It is the inverse of RESET.
OSC SEL	-	8	When OSC SEL is unconnected or driven high, the internal oscillator sets the reset time delay and watchdog timeout period. When OSC SEL is low, the external oscillator input, OSC IN, is enabled. OSC SEL has a 3μA internal pullup. See Table 1.
OSC IN	-	7	OSC IN sets the Reset delay timing and Watchdog timeout period when OSC SEL floats or is driven high. The timing can also be adjusted by connecting an external capacitor to this pin. See Figure 8. When OSC SEL is low, OSC IN selects between fast and slow Watchdog timeout periods.
WDO	-	14	The Watchdog Output, WDO, goes low if WDI remains either high or low for longer than the Watchdog timeout period. WDO is set high by the next transition at WDI. If WDI is unconnected or at mid-supply, WDO remains high. WDO also goes high when LOW LINE goes low.

Microprocessor Supervisory Circuits

MAX690/91/92/93

Typical Applications

MAX691 and MAX693

A typical connection for the MAX691/693 is shown in Figure 1. CMOS RAM is powered from V_{OUT}. V_{OUT} is internally connected to V_{CC} when 5V power is present, or to V_{BATT} when V_{CC} is less than the battery voltage. V_{OUT} can supply 100mA from V_{CC}, but if more current is required, an external PNP transistor can be added. When V_{CC} is higher than V_{BATT}, the BATT ON output goes low, providing 7mA of base drive for the external transistor. When V_{CC} is lower than V_{BATT}, an internal 500Ω MOSFET connects the backup battery to V_{OUT}. The quiescent current in the battery backup mode is 1μA maximum when V_{CC} is between 0V and V_{BATT} - 700mV.

Reset Output

A voltage detector monitors V_{CC} and generates a RESET output to hold the microprocessor's Reset line low when V_{CC} is below 4.65V (4.4V for MAX693). An internal monostable holds RESET low for 50ms after V_{CC} rises above 4.65V (4.4V for MAX693). This prevents repeated toggling of RESET even if the 5V power drops out and recovers with each power line cycle.

The crystal oscillator normally used to generate the clock for microprocessors takes several milliseconds to start. Since most microprocessors need several clock cycles to reset, RESET must be held low until the microprocessor clock oscillator has started. The

MAX690 Family power-up RESET pulse lasts 50ms to allow for this oscillator start-up time. The manual reset switch and the 0.1μF capacitor connected to the reset bus can be omitted if manual reset is not needed. An inverted, active high, RESET output is also supplied.

Power Fail Detector

The MAX691/93 issues a non-maskable interrupt (NMI) to the microprocessor when a power failure occurs. The +5V power line is monitored via two external resistors connected to the Power Fail Input (PFI). When the voltage at PFI falls below 1.25V, the Power Fail Output (PFO) drives the processor's NMI input low. If a Power Fail threshold of 4.8V is chosen, the microprocessor will have the time when V_{CC} falls from 4.8V to 4.65V to save data into RAM. An earlier power fail warning can be generated if the unregulated DC input of the 5V regulator is available for monitoring.

RAM Write Protection

The MAX691/93 CE OUT line drives the Chip Select inputs of the CMOS RAM. CE OUT follows CE IN as long as V_{CC} is above the 4.65V (4.4V for MAX693) reset threshold. If V_{CC} falls below the reset threshold, CE OUT goes high, independent of the logic level at CE IN. This prevents the microprocessor from writing erroneous data into RAM during power-up, power-down, brownouts, and momentary power interruptions. The LOW LINE output goes low when V_{CC} falls below 4.65V (4.4V for MAX693).

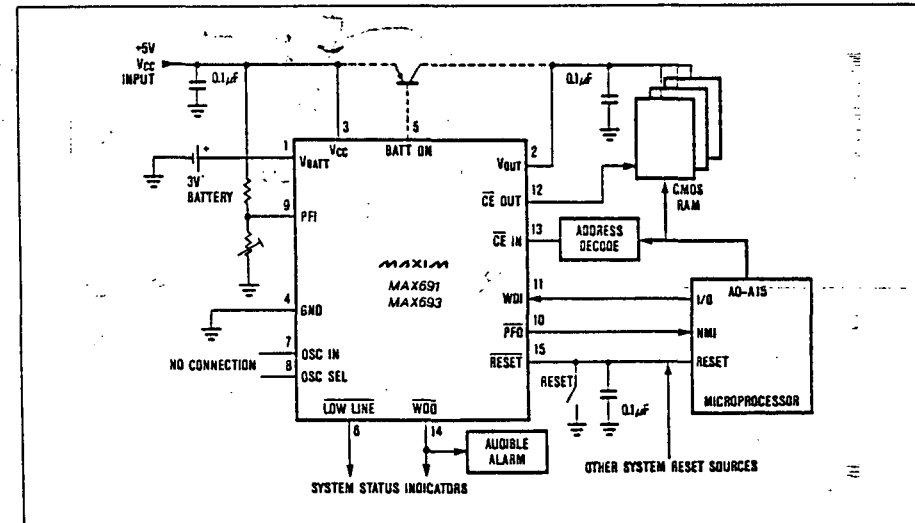


Figure 1 MAX691/693 Typical Application

Microprocessor Supervisory Circuits

Watchdog Timer

The microprocessor drives the WATCHDOG INPUT (WDI) with an I/O line. When OSC IN and OSC SEL are unconnected, the microprocessor must toggle the WDI pin once every 1.6 seconds to verify proper software execution. If a hardware or software failure occurs such that WDI is not toggled, the MAX691/93 will issue a 50ms RESET pulse after 1.6 seconds. This typically restarts the microprocessor's power-up routine. A new RESET pulse is issued every 1.6 seconds until WDI is again strobed.

The WATCHDOG OUTPUT (WDO) goes low if the watchdog timer is not serviced within its timeout period. Once WDO goes low it remains low until a transition occurs at WDI. The watchdog timer feature can be disabled by leaving WDI unconnected. OSC IN and OSC SEL also allow other watchdog timing options, as shown in Table 1 and Figure 8.

MAX690 and MAX692

The 8-pin MAX690 and MAX692 have most of the

features of the MAX691 and MAX693. Figure 2 shows the MAX690/692 in a typical application. Operation is much the same as with the MAX691/693 (Figure 1) but in this case the Power Fail Input (PFI) monitors the unregulated input to the 7805 regulator. The MAX690 RESET output goes low when V_{CC} falls below 4.65V. The RESET output of the MAX692 goes low when V_{CC} drops below 4.4V.

The current consumption of the battery-backed-up power bus must be less than 100mA. The MAX690/692 does not have a BATT ON output to drive an external transistor. The MAX 690/92 also does not include chip enable gating circuitry that is available on the MAX 691/93. In many systems though, CE gating is not needed since a low input to the microprocessor RESET line prevents the processor from writing to RAM during power-up and power-down transients.

The MAX690/92 watchdog timer has a fixed 1.6 second timeout period. If WDI remains either low or high for more than 1.6 seconds, a RESET pulse is sent to the microprocessor. The watchdog timer is disabled if WDI is left floating.

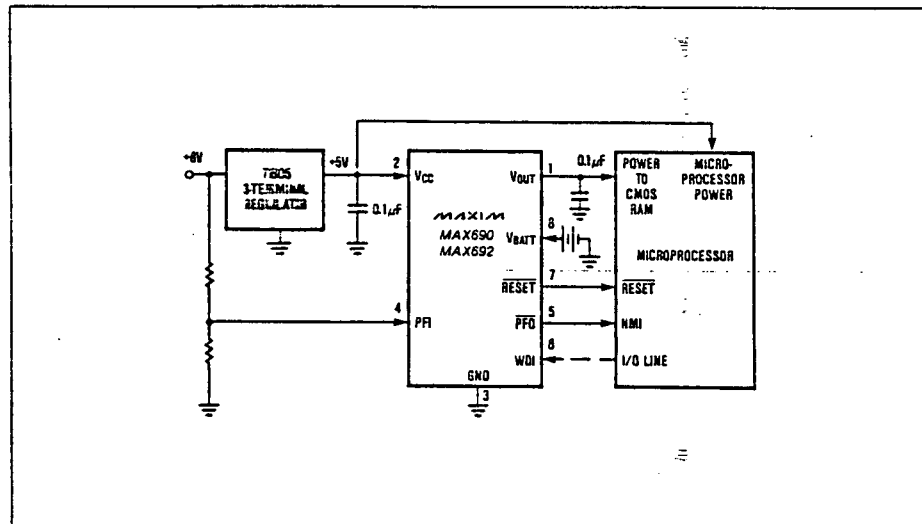


Figure 2. MAX690/692 Typical Application

Microprocessor Supervisory Circuits

Detailed Description

Battery-Switchover and V_{OUT}

The battery switchover circuit compares V_{CC} to the V_{BATT} input, and connects V_{OUT} to whichever is higher. Switchover occurs when V_{CC} is 50mV greater than V_{BATT} as V_{CC} falls, and when V_{CC} is 70mV more than V_{BATT} as V_{CC} rises (See Figure 4). The switchover comparator has 20mV of hysteresis to prevent repeated, rapid switching if V_{CC} falls very slowly or remains nearly equal to the battery voltage.

When V_{CC} is higher than V_{BATT} , V_{CC} is internally switched to V_{OUT} via a low saturation PNP transistor. V_{OUT} has 100mA output current capability and thermal shutdown short circuit protection. Use an external PNP pass transistor in parallel with the internal transistor if the output current requirement at V_{OUT} exceeds 100mA or if a lower V_{CC} - V_{OUT} voltage differential is desired. The BATT ON output (MAX691/693 only) can directly drive the base of the external transistor.

It should be noted that the MAX690/91/92/93 need only supply the average current drawn by the CMOS RAM if there is adequate filtering. Many RAM data sheets specify a 75mA maximum supply current, but this peak current spike lasts only 100ns. A 0.1µF bypass capacitor at V_{OUT} supplies the high instantaneous current, while V_{OUT} need only supply the average load current, which is much less. A capacitance of 0.1µF or greater must be connected to the V_{OUT} terminal to ensure stability.

A 500 ohm MOSFET connects the V_{BATT} input to V_{OUT}

during battery backup. This MOSFET has very low input-to-output differential (dropout voltage) at the low current levels required for battery backup of CMOS RAM or other low power CMOS circuitry. When V_{CC} equals V_{BATT} the supply current is typically 12µA. When V_{CC} is between 0V and (V_{BATT} - 700mV) the typical supply current is only 600nA typical, 1µA maximum.

The MAX690/691 operates with battery voltages from 2.0V to 4.25V while the MAX692/693 operates with battery voltages from 2.0V to 4.0V. High value capacitors, either standard electrolytic or the farad-size double layer capacitors, can also be used for short-term memory backup. The charging resistor for both capacitors and rechargeable batteries should be connected to V_{OUT} since this eliminates the discharge path that exists if the resistor is connected to V_{CC} .

A small charging current of typically 10nA (5µ max) flows out of the V_{BATT} terminal. This current varies with the amount of current that is drawn from V_{OUT} but its polarity is such that the backup battery is always slightly charged, and is never discharged while V_{CC} is in its operating voltage range. This extends the shelf life of the backup battery by compensating for its self-discharge current. Also note that this current poses no problem when lithium batteries are used for backup since the maximum charging current (5µA) is safe for even the smallest lithium cells.

If the battery-switchover section is not used, connect V_{BATT} to GND and connect V_{OUT} to V_{CC} . Table 2 shows the state of the inputs and output in the low power battery backup mode.

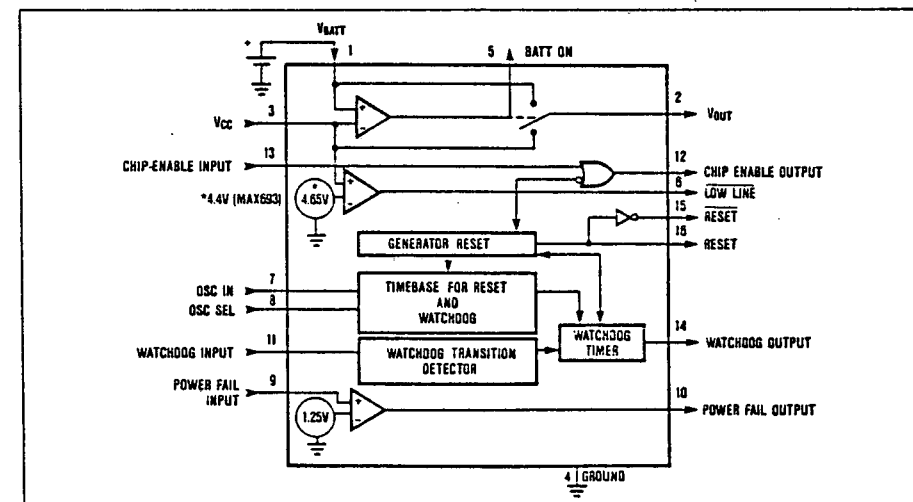


Figure 3. MAX691/693 Block Diagram

Microprocessor Supervisory Circuits

MAX690/91/92/93

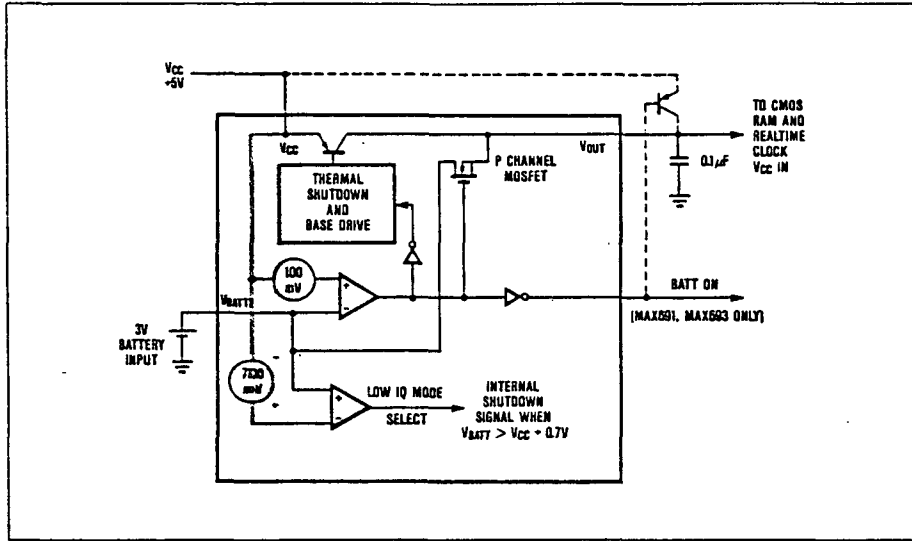


Figure 4 Battery-Switchover Block Diagram

Reset Output

RESET is an active low output which goes low whenever V_{CC} falls below 4.5V (MAX690/691) or 4.25V (MAX692/693). It will remain low until V_{CC} rises above 4.75V (MAX690/691) or 4.5V (MAX692/693) for 50 milliseconds. (See Figures 5 and 6.)

The guaranteed minimum and maximum thresholds of the MAX690/691 are 4.5V and 4.75V, while the guaranteed thresholds of the MAX692/693 are 4.25V and 4.5V. The MAX690/691 is compatible with 5V supplies with a +10%, -5% tolerance while the MAX692/693 is compatible with 5V ±10% supplies. The reset threshold comparator has approximately 50mV of hysteresis, with a nominal threshold of 4.65V in the MAX690/691, and 4.4V in the MAX692/693.

The response time of the reset voltage comparator is about 100µs. V_{CC} should be bypassed to ensure that glitches do not activate the **RESET** output.

RESET also goes low if the Watchdog Timer is enabled and WDI remains either high or low longer than the watchdog timeout period. **RESET** has an internal 3µA pullup, and can either connect to an open collector Reset bus or directly drive a CMOS gate without an external pullup resistor.

CE Gating and RAM Write Protection

The MAX691 and MAX693 use two pins to control the Chip Enable or Write inputs of CMOS RAMs. When V_{CC} is +5V, **CE OUT** is a buffered replica of **CE IN**, with a 50ns propagation delay. If V_{CC} input falls below 4.65V (4.5V min, 4.75V max) an internal gate forces **CE OUT** high, independent of **CE IN**. The MAX693 **CE OUT** goes high whenever V_{CC} is below 4.4V (4.25V min, 4.5V max). The **CE** output of both devices is also forced high when V_{CC} is less than V_{BATT} . (See Figure 5.)

CE OUT typically drives the **CE**, **CS**, or **Write** input of battery backed up CMOS RAM. This ensures the integrity of the data in memory by preventing write operations when V_{CC} is at an invalid level. Similar protection of EEPROMs can be achieved by using the **CE OUT** to drive the Store or Write inputs of an EEPROM, EAROM, or NOVRAM.

If the 50ns typical propagation delay of **CE OUT** is too long, connect **CE IN** to GND and use the resulting **CE OUT** to control a high speed external logic gate. A second alternative is to AND the **LOW LINE** output with the **CE** or **WR** signal. An external logic gate and the **RESET** output of the MAX690/692 can also be used for CMOS RAM write protection.

Microprocessor Supervisory Circuits

MAX690/91/92/93

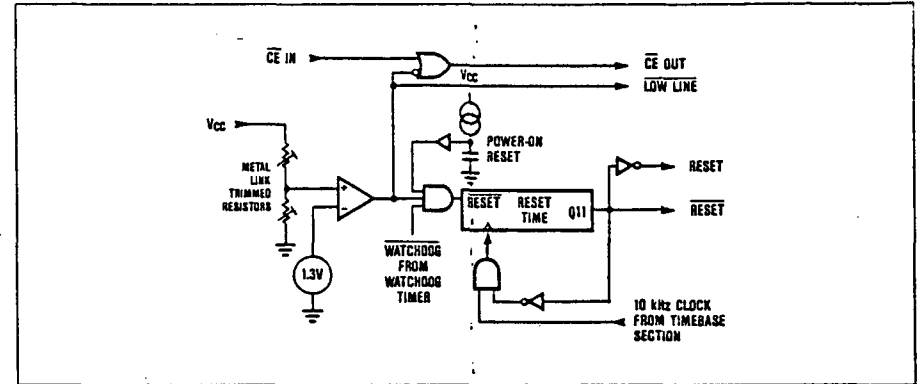


Figure 5. Reset Block Diagram

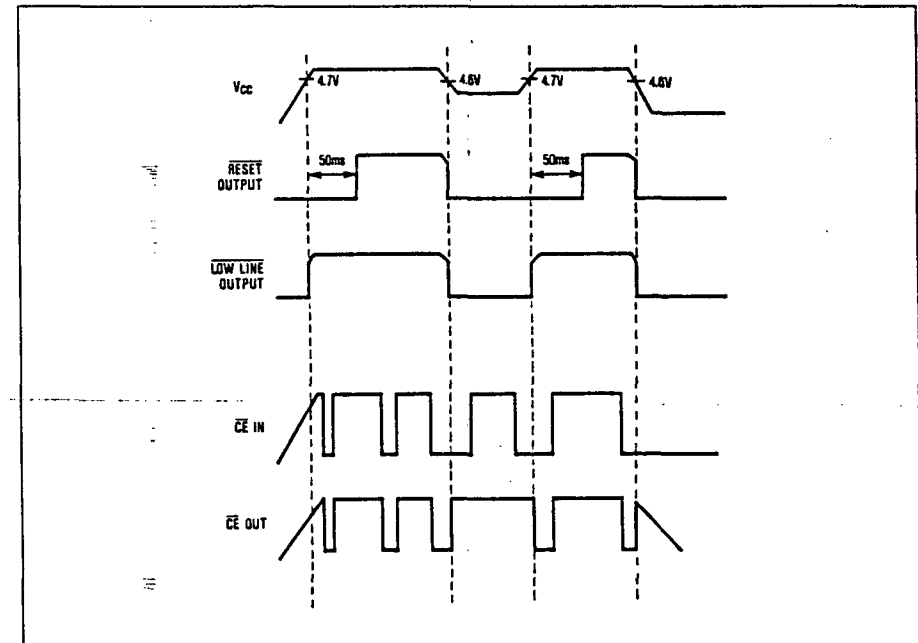


Figure 6. MAX691 Reset Timing

Microprocessor Supervisory Circuits

1.25V Comparator and Power Fail Warning

The Power Fail Input (PFI) is compared to an internal 1.25V reference. The Power Fail Output (PFO) goes low when the voltage at PFI is less than 1.25V. Typically PFI is driven by an external voltage divider which senses either the unregulated DC input to the system's 5V regulator or the regulated 5V output. The voltage divider ratio can be chosen such that the voltage at PFI falls below 1.25V several milliseconds before the +5V supply falls below 4.75V. PFO is normally used to interrupt the microprocessor so that data can be stored in RAM before V_{CC} falls below 4.75V and the RESET output goes low (4.5V for MAX692/93).

The Power Fail Detector can also monitor the backup battery to warn of a low battery condition. To conserve battery power, the Power Fail Detector comparator is turned off and PFO is forced low when V_{CC} is lower than the V_{BATT} input voltage.

Watchdog Timer and Oscillator

The watchdog circuit monitors the activity of the microprocessor. If the microprocessor does not toggle the Watchdog Input (WDI) within the selected timeout period, a 50 millisecond RESET pulse is generated. Since many systems cannot service the watchdog timer immediately after a reset, the MAX691/693 has a longer timeout period after a reset is issued. The normal timeout period becomes effective following the first

transition of WDI after RESET has gone high. The watchdog timer is restarted at the end of Reset, whether the Reset was caused by lack of activity on WDI or by V_{CC} falling below the reset threshold. If WDI remains either high or low, reset pulses will be issued every 1.6 seconds. The watchdog monitor can be deactivated by floating the Watchdog Input (WDI).

The Watchdog Output (WDO, MAX691/693 only) goes low if the watchdog timer "times out", and it remains low until set high by the next transition on the watchdog input. WDO is also set high when V_{CC} goes below the reset threshold.

The watchdog timeout period is fixed at 1.6 seconds and the reset pulse width is fixed at 50ms on the 8-pin MAX690 and MAX692. The MAX691 and MAX693 allow these times to be adjusted per Table 1. Figure 8 shows various oscillator configurations.

The internal oscillator is enabled when OSC SEL is high or floating. In this mode, OSC IN selects between the 1.6 second and 100ms watchdog timeout periods. In either case, immediately after a reset the timeout period is 1.6 seconds. This gives the microprocessor time to reinitialize the system. If OSC IN is low, then the 100ms watchdog period becomes effective after the first transition of WDI. The software should be written such that the I/O port driving WDI is left in its power-up reset state until the initialization routines are completed and the microprocessor is able to toggle WDI at the minimum watchdog timeout period of 70ms.

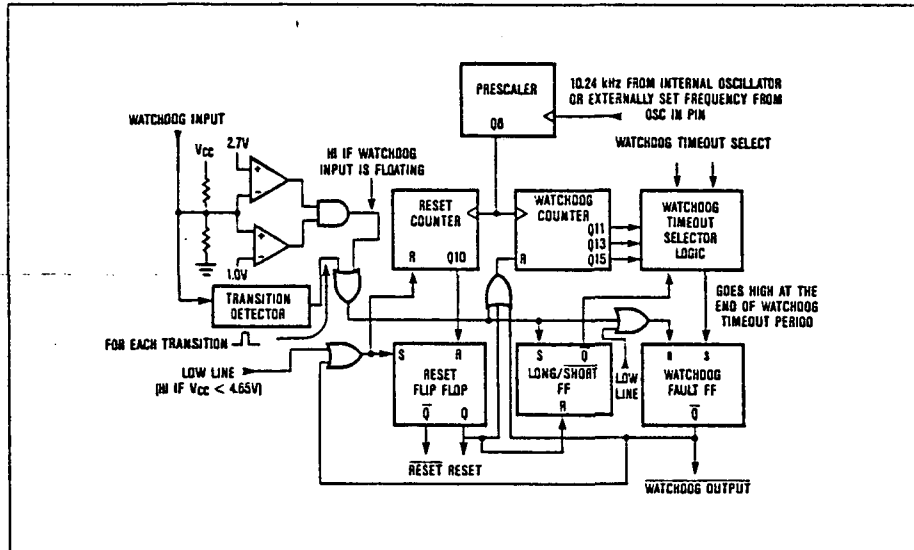


Figure 7. Watchdog Timer Block Diagram

Microprocessor Supervisory Circuits

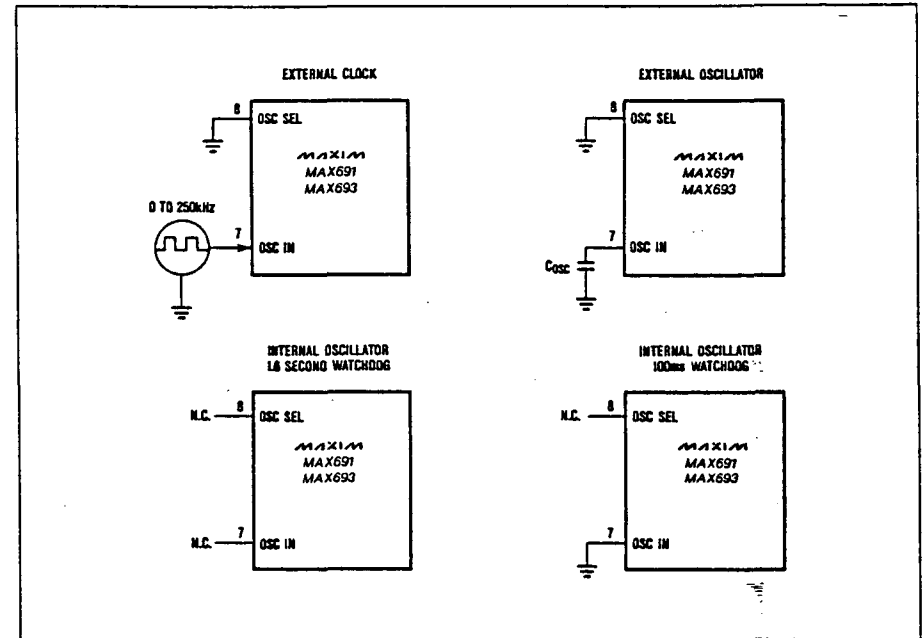


Figure 8. Oscillator Circuits

Table 1. MAX691 and MAX693 Reset Pulse Width and Watchdog Timeout Selections

OSC SEL	OSC IN	Watchdog Timeout Period		Reset Timeout Period
		Normal	Immediately After Reset	
Low	External Clock input	1024 clks	4096 clks	512 clks
Low	External Capacitor	$\frac{400\text{ms}}{47\text{pF}} \times C$	$\frac{1.6\text{ sec}}{47\text{pF}} \times C$	$\frac{200\text{ms}}{47\text{pF}} \times C$
High/Floating	Low	100ms	1.6 sec	50ms
High/Floating	High / Floating	1.6 sec	1.6 sec	50ms

Note 1. The MAX690 watchdog timeout period is fixed at 1.6 seconds nominal, the MAX690 Reset pulse width is fixed at 50ms nominal.
 Note 2. When the MAX691 OSC SEL pin is low, OSC IN can be driven by an external clock signal, or an external capacitor can be connected between OSC IN and GND. The nominal internal oscillator frequency is 10.24kHz. The nominal oscillator frequency with external capacitor is
$$F_{\text{osc}} = \frac{1.75 \times 10^7}{C_{\text{osc}}}$$
 (Hz) (Farads)

Note 3. See Electrical Specifications Table for minimum and maximum timing values.

Microprocessor Supervisory Circuits

Application Hints

Other Uses of the Power Fail Detector

In Figure 9 the Power Fail Detector is used to initiate a system reset when V_{CC} falls to 4.85V. Since the threshold of the Power Fail Detector is not as accurate as the onboard Reset voltage detector, a trimpot must be used to adjust the voltage detection threshold. Both the PFO and RESET outputs have high sink current capability and only 10 μ A of source current drive. This allows the two outputs to be connected directly to each other in a "wired or" fashion.

The overvoltage detector circuit in Figure 10 resets the microprocessor whenever the nominal 5V V_{CC} is above 5.5V. The battery monitor circuit (Figure 11) shows the status of the memory backup battery. If desired, the CE OUT can be used to apply a test load to the battery. Since CE OUT is forced high during the battery backup mode, the test load will not be applied to the battery while it is in use, even if the microprocessor is not powered.

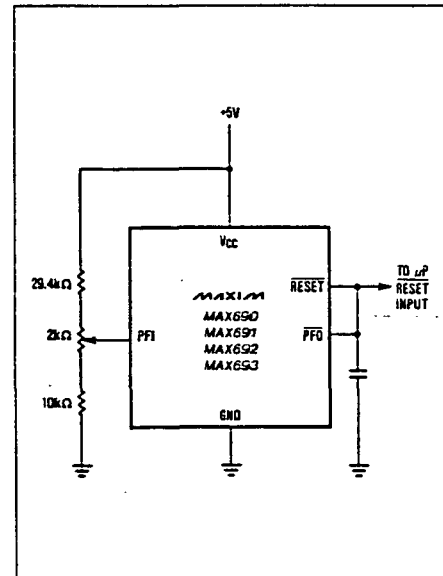


Figure 9. Externally Adjustable V_{CC} Reset Threshold

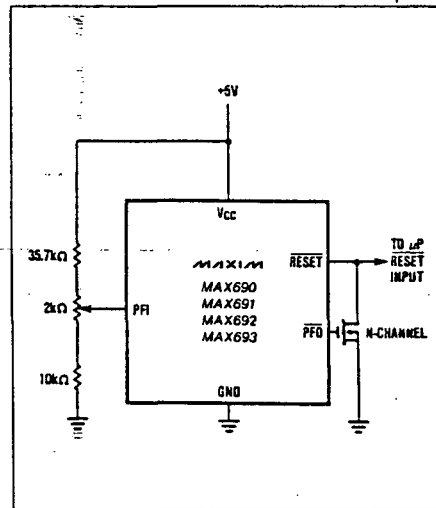


Figure 10. Reset on Overvoltage or Undervoltage

Adding Hysteresis to the Power Fail Comparator

Since the power fail comparator circuit is non-inverting, hysteresis can be added by connecting a resistor between the PFO output and the PFI input as shown in Figure 12. When PFO is low, resistor R3 sinks current from the summing junction at the PFI pin. When PFO is high, the series combination of R3 and R4 source current into the PFI summing junction.

Alternate Watchdog Input Drive Circuits

The Watchdog feature can be enabled and disabled under program control by driving WDI with a 3-state buffer (Figure 13). The drawback to this circuit is that a software fault may erroneously 3-state the buffer, thereby preventing the MAX690 from detecting that the microprocessor is no longer working. In most cases a better method is to extend the watchdog period rather than disabling the watchdog. See Figure 14. When the control input is high, the OSC SEL pin is low and the watchdog timeout is set by the external capacitor. A 0.01 μ F capacitor sets a watchdog timeout delay of 100 seconds. When the control input is low, the OSC SEL pin is driven high, selecting the internal oscillator. The 100ms or the 1.6 sec period is chosen, depending on which diode in Figure 14 is used.

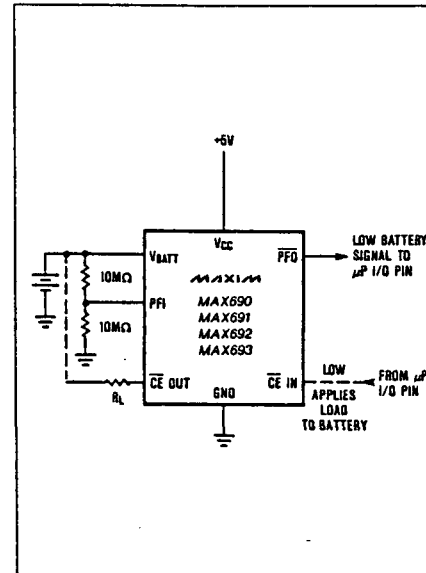


Figure 11. Backup Battery Monitor with Optional Test Load

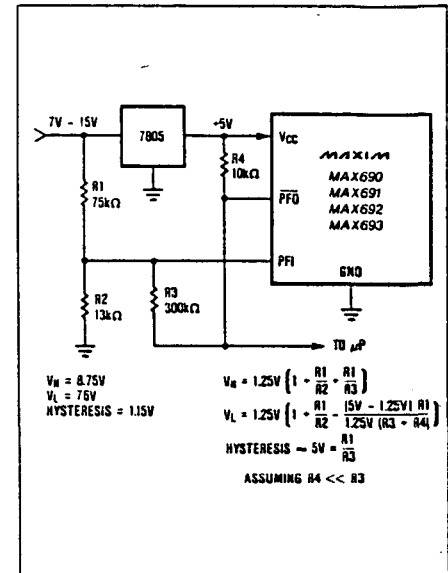


Figure 12. Adding hysteresis to the Power Fail Voltage Comparator

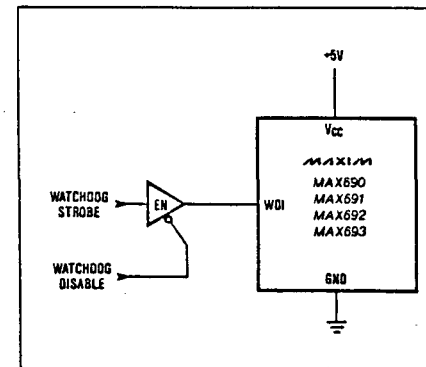


Figure 13. Disabling the Watchdog Under Program Control

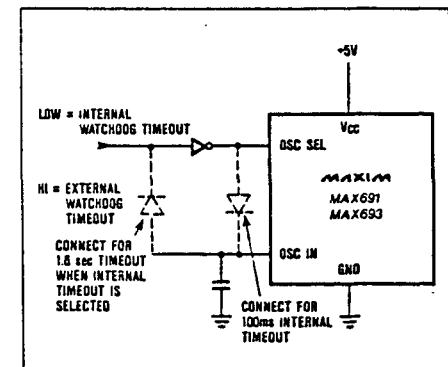


Figure 14. Selecting internal or External Watchdog Timeout

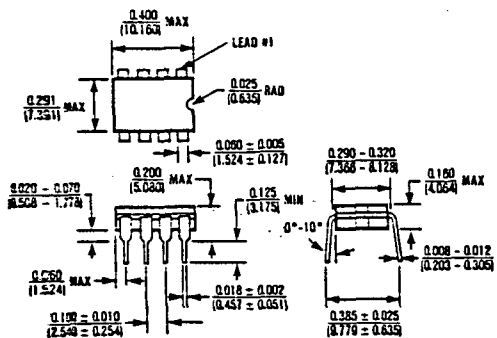
Microprocessor Supervisory Circuits

Microprocessor Supervisory Circuits

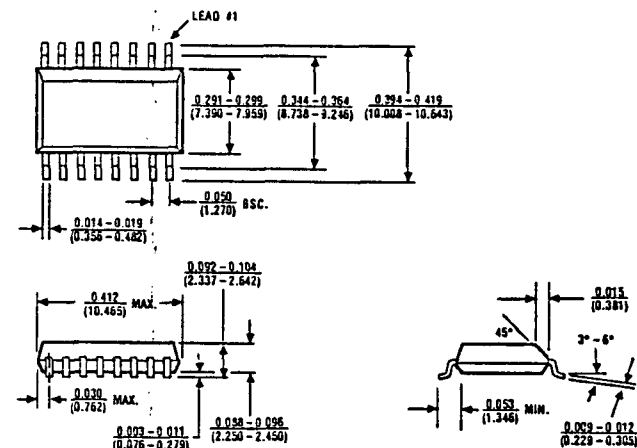
Table 2. Input and Output Status In Battery Backup Mode

V_{BATT} , V_{OUT}	V_{BATT} is connected to V_{OUT} via internal MOSFET.
RESET	logic low
RESET	logic high. The open circuit output voltage is equal to V_{OUT} .
LOW LINE	logic low
BATT ON	logic high
WDI	WDI is internally disconnected from its internal pullup and does not source or sink current as long as its input voltage is between GND and V_{OUT} . The input voltage does not affect supply current.
WDO	logic high
PFI	The Power Fail Comparator is turned off and the Power Fail Input voltage has no effect on the Power Fail Output.
PFO	logic low
CE IN	CE IN has a $2\mu A$ input pullup current source. Float or drive high to minimize supply current.
CE OUT	logic high.
OSC IN	OSC IN is ignored.
OSC SEL	OSC SEL is ignored.
V_{CC}	Approximately $12\mu A$ is drawn from the V_{BATT} input when V_{CC} is between $V_{BATT} + 100mV$ and $V_{BATT} - 700mV$. The supply current is $1\mu A$ maximum when V_{CC} is less than $V_{BATT} - 700mV$.

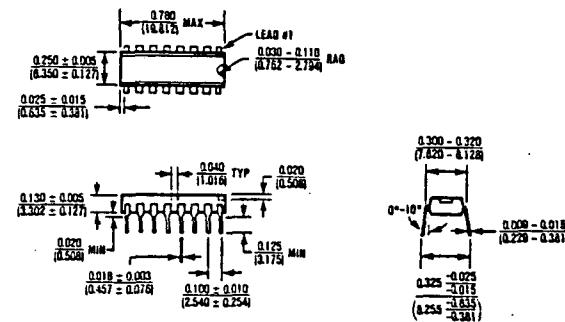
Package Information



8 Lead CERDIP (JA)
 $\theta_{JA} = 125^{\circ}C/W$
 $\theta_{JC} = 55^{\circ}C/W$



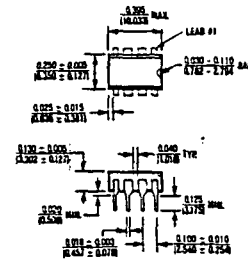
16 Lead Small Outline, Wide (WE)
 $\theta_{JA} = 105^{\circ}C/W$
 $\theta_{JC} = 60^{\circ}C/W$



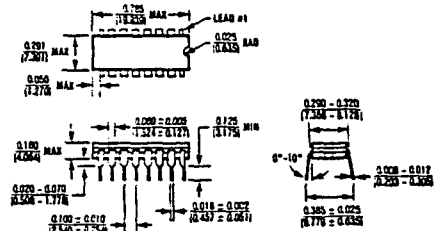
16 Lead Plastic DIP (PE)
 $\theta_{JA} = 100^{\circ}C/W$
 $\theta_{JC} = 60^{\circ}C/W$

Microprocessor Supervisory Circuits

MAX690/91/92/93



8 Lead Plastic DIP (PA)
 $\theta_{JA} = 120^{\circ}\text{C/W}$
 $\theta_{JC} = 70^{\circ}\text{C/W}$

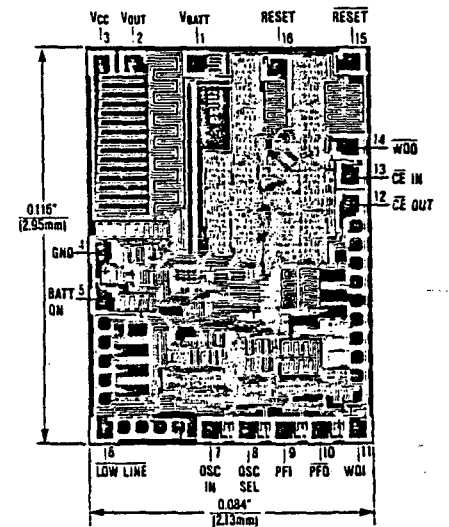


16 Lead Cerdip (JE)
 $\theta_{JA} = 100^{\circ}\text{C/W}$
 $\theta_{JC} = 50^{\circ}\text{C/W}$

Ordering Information

PART	TEMP. RANGE	PACKAGE
MAX692CPA	0°C to +70°C	8 Lead Plastic DIP
MAX692EPA	-40°C to +85°C	8 Lead Plastic DIP
MAX692EJA	-40°C to +85°C	8 Lead Cerdip
MAX692MJA	-55°C to +125°C	8 Lead Cerdip
MAX693C/D	0°C to +70°C	Dice
MAX693CPE	0°C to +70°C	16 Lead Plastic DIP
MAX693CWE	0°C to +70°C	16 Lead Small Outline
MAX693EPE	-40°C to +85°C	16 Lead Plastic DIP
MAX693EJE	-40°C to +85°C	16 Lead Cerdip
MAX693EWE	-40°C to +85°C	16 Lead Small Outline
MAX693MJE	-55°C to +125°C	16 Lead Cerdip

Chip Topography



Maxim cannot assume responsibility for use of any circuitry other than circuitry entirely embodied in a Maxim product. No circuit patent licenses are implied. Maxim reserves the right to change the circuitry and specifications without notice at any time.

16 Maxim Integrated Products, 510 N. Pastoria Avenue, Sunnyvale, CA 94086 (408) 737-7600

ADC0808, ADC0809 8-Bit μ P Compatible A/D Converters With 8-Channel Multiplexer

General Description

The ADC0808, ADC0809 data acquisition component is a monolithic CMOS device with an 8-bit analog-to-digital converter, 8-channel multiplexer and microprocessor compatible control logic. The 8-bit A/D converter uses successive approximation as the conversion technique. The converter features a high impedance chopper stabilized comparator, a 256R voltage divider with analog switch tree and a successive approximation register. The 8-channel multiplexer can directly access any of 8 single-ended analog signals.

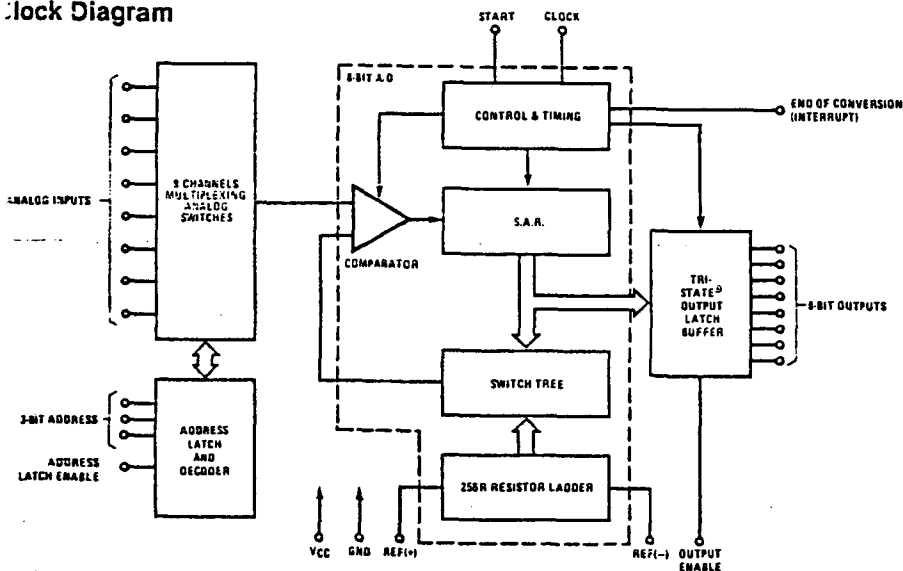
The device eliminates the need for external zero and full-scale adjustments. Easy interfacing to microprocessors is provided by the latched and decoded multiplexer address inputs and latched TTL TRI-STATE[®] outputs.

The design of the ADC0808, ADC0809 has been optimized incorporating the most desirable aspects of several conversion techniques. The ADC0808, ADC0809 offers high speed, high accuracy, minimal temperature dependence, excellent long-term accuracy and repeatability, and consumes minimal power. These features make the device ideally suited to applications from process and machine control to consumer and automotive applications. For 16-channel multiplexer with common output (multiplexed port) see ADC0816 data sheet. (See AN-247 for more information.)

Features

- Resolution — 8-bits
- Total unadjusted error — $\pm 1/2$ LSB and ± 1 LSB
- No missing codes
- Conversion time — 100 μ s
- Single supply — 5 V_{CC}
- Operates ratiometrically or with 5 V_{DC} or analog span adjusted voltage reference
- 8-channel multiplexer with latched control logic
- Easy interface to all microprocessors, or operates "stand alone"
- Outputs meet TTL voltage level specifications
- 0V to 5V analog input voltage range with single 5V supply
- No zero or full-scale adjust required
- Standard hermetic or molded 28-pin DIP package
- Temperature range -40°C to $+85^{\circ}\text{C}$ or -55°C to $+125^{\circ}\text{C}$
- Low power consumption — 15 mW
- Latched TRI-STATE[®] output

Block Diagram



Absolute Maximum Ratings (Notes 1 and 2)

Supply Voltage (V_{CC}) (Note 3)	6.5V
Voltage at Any Pin Except Control Inputs	$-0.3\text{V to } (V_{CC} + 0.3\text{V})$
Voltage at Control Inputs (START, OE, CLOCK, ALE, ADD A, ADD B, ADD C)	$-0.3\text{V to } +15\text{V}$
Storage Temperature Range	$-65^{\circ}\text{C to } +150^{\circ}\text{C}$
Package Dissipation at $T_A = 25^{\circ}\text{C}$	875 mW
Lead Temperature (Soldering, 10 seconds)	300 $^{\circ}\text{C}$

Operating Ratings (Notes 1 and 2)

Temperature Range (Note 1)	$T_{MIN} \leq T_A \leq T_{MAX}$ $-55^{\circ}\text{C} \leq T_A \leq +125^{\circ}\text{C}$
ADC0808CJ ADC0808CCJ, ADC0808CCN, ADC0809CCN	$-40^{\circ}\text{C} \leq T_A \leq +85^{\circ}\text{C}$
Range of V_{CC} (Note 1)	$4.5\text{V} \leq V_{CC} \leq 5.5\text{V}$

Electrical Characteristics

Converter Specifications: $V_{CC} = 5\text{V}$, $V_{DC} = V_{REF(+)}$, $V_{REF(-)} = \text{GND}$, $T_{MIN} \leq T_A \leq T_{MAX}$ and $f_{CLK} = 640\text{ kHz}$ unless otherwise stated.

Parameter	Conditions	Min	Typ	Max	Units
ADC0808 Total Unadjusted Error (Note 5)	25°C T_{MIN} to T_{MAX}			$\pm 1/2$ $\pm 3/4$	LSB LSB
ADC0809 Total Unadjusted Error (Note 5)	$0^{\circ}\text{C to } 70^{\circ}\text{C}$ T_{MIN} to T_{MAX}			± 1 $\pm 1 1/4$	LSB LSB
Input Resistance	From Ref (+) to Ref (-)	1.0	2.5		k Ω
Analog Input Voltage Range	(Note 4) V (+) or V (-)	GND-0.10		$V_{CC} + 0.10$	V_{DC}
$V_{REF(+)}$ Voltage, Top of Ladder	Measured at Ref (+)		V_{CC}	$V_{CC} + 0.1$	V
$\frac{V_{REF(+)} + V_{REF(-)}}{2}$ Voltage, Center of Ladder		$V_{CC}/2 - 0.1$	$V_{CC}/2$	$V_{CC}/2 + 0.1$	V
$V_{REF(-)}$ Voltage, Bottom of Ladder	Measured at Ref (-)	-0.1	0		V
Comparator Input Current	$f_c = 640\text{ kHz}$, (Note 6)	-2	± 0.5	2	μA

Electrical Characteristics

Digital Levels and DC Specifications: ADC0808CJ $4.5\text{V} \leq V_{CC} \leq 5.5\text{V}$, $-55^{\circ}\text{C} \leq T_A \leq +125^{\circ}\text{C}$ unless otherwise noted
ADC0808CCJ, ADC0808CCN, and ADC0809CCN $4.75 \leq V_{CC} \leq 5.25\text{V}$, $-40^{\circ}\text{C} \leq T_A \leq +85^{\circ}\text{C}$ unless otherwise noted

Parameter	Conditions	Min	Typ	Max	Units
ANALOG MULTIPLEXER					
$I_{OFF(+)}$	OFF Channel Leakage Current $V_{CC} = 5\text{V}$, $V_{IN} = 5\text{V}$, $T_A = 25^{\circ}\text{C}$ T_{MIN} to T_{MAX}		10	200	nA μA
$I_{OFF(-)}$	OFF Channel Leakage Current $V_{CC} = 5\text{V}$, $V_{IN} = 0$, $T_A = 25^{\circ}\text{C}$ T_{MIN} to T_{MAX}	-200 -1.0	-10		nA μA
CONTROL INPUTS					
$V_{IN(1)}$	Logical "1" Input Voltage		$V_{CC} - 1.5$		V
$V_{IN(0)}$	Logical "0" Input Voltage			1.5	V
$I_{IN(1)}$	Logical "1" Input Current (The Control Inputs) $V_{IN} = 15\text{V}$			1.0	μA
$I_{IN(0)}$	Logical "0" Input Current (The Control Inputs) $V_{IN} = 0$	-1.0			μA
I_{CC}	Supply Current $f_{CLK} = 640\text{ kHz}$		0.3	3.0	mA





Electrical Characteristics (Continued)

Digital Levels and DC Specifications: ADC0808CJ 4.5V $\leq V_{CC} \leq 5.5V$, $-55^{\circ}C \leq T_A \leq +125^{\circ}C$ unless otherwise noted
 ADC0808CCJ, ADC0808CCN, and ADC0809CCN 4.75 $\leq V_{CC} \leq 5.25V$, $-40^{\circ}C \leq T_A \leq +85^{\circ}C$ unless otherwise noted

Parameter	Conditions	Min	Typ	Max	Units
DATA OUTPUTS AND EOC (INTERRUPT)					
$V_{OUT(1)}$	Logical "1" Output Voltage	$I_O = -30 \mu A$	$V_{CC}-0.4$		V
$V_{OUT(0)}$	Logical "0" Output Voltage	$I_O = 1.6 \text{ mA}$		0.45	V
$V_{OUT(EOC)}$	Logical "0" Output Voltage EOC	$I_O = 1.2 \text{ mA}$		0.45	V
I_{OUT}	TRI-STATE [®] Output Current	$V_O = 5V$ $V_O = 0$	-3	3	μA μA

Electrical Characteristics

Timing Specifications: $V_{CC} = V_{REF(+)} = 5V$, $V_{REF(-)} = GND$, $t_r = t_f = 20 \text{ ns}$ and $T_A = 25^{\circ}C$ unless otherwise noted.

Symbol	Parameter	Conditions	Min	Typ	Max	Units
t_{WS}	Minimum Start Pulse Width	(Figure 5)		100	200	ns
t_{WALE}	Minimum ALE Pulse Width	(Figure 5)		100	200	ns
t_s	Minimum Address Set-Up Time	(Figure 5)		25	50	ns
t_H	Minimum Address Hold Time	(Figure 5)		25	50	ns
t_D	Analog MUX Delay Time From ALE	$R_S = 0\Omega$ (Figure 5)		1	2.5	μs
t_{HI}, t_{HO}	OE Control to Q Logic State	$C_L = 50 \text{ pF}$, $R_L = 10k$ (Figure 8)		125	250	ns
t_{TH}, t_{OH}	OE Control to Hi-Z	$C_L = 10 \text{ pF}$, $R_L = 10k$ (Figure 8)		125	250	ns
t_c	Conversion Time	$f_c = 640 \text{ kHz}$, (Figure 5) (Note 7)	90	100	116	μs
f_c	Clock Frequency		10	640	1280	kHz
t_{EOC}	EOC Delay Time	(Figure 5)	0		$8 + 2 \mu s$	Clock Periods
C_{IN}	Input Capacitance	At Control Inputs		10	15	pF
C_{OUT}	TRI-STATE [®] Output Capacitance	At TRI-STATE [®] Outputs, (Note 12)		10	15	pF

- Note 1: Absolute maximum ratings are those values beyond which the life of the device may be impaired.
- Note 2: All voltages are measured with respect to GND, unless otherwise specified.
- Note 3: A zener diode exists, internally, from V_{CC} to GND and has a typical breakdown voltage of 7 V_{DC} .
- Note 4: Two on-chip diodes are tied to each analog input which will forward conduct for analog input voltages one diode drop below ground or one diode drop greater than the V_{CC} supply. The spec allows 100 mV forward bias of either diode. This means that as long as the analog V_{IN} does not exceed the supply voltage by more than 100 mV, the output code will be correct. To achieve an absolute 0 V_{DC} to 5 V_{DC} input voltage range will therefore require a minimum supply voltage of 4.500 V_{DC} over temperature variations, input tolerance and loading.
- Note 5: Total unadjusted error includes offset, half-scale, linearity, and multiplexer errors. See Figure 3. None of these A/Ds requires a zero or full-scale adjust. However, if an all zero code is desired for an analog input other than 0.0V, or if a narrow full-scale span exists (for example: 0.5V to 4.5V full-scale) the reference voltages can be adjusted to achieve this. See Figure 13.
- Note 6: Comparator input current is a bias current into or out of the chopper stabilized comparator. The bias current varies directly with clock frequency and has little temperature dependence (Figure 6). See paragraph 4.0.
- Note 7: The outputs of the data register are updated one clock cycle before the rising edge of EOC.

Functional Description

Multiplexer: The device contains an 8-channel single-ended analog multiplexer. A particular input channel is selected by using the address decoder. Table I shows the input states for the address lines to select any channel. The address is latched into the decoder on the low-to-high transition of the address latch enable signal.

TABLE I

SELECTED ANALOG CHANNEL	ADDRESS LINE		
	C	B	A
IN0	L	L	L
IN1	L	L	H
IN2	L	H	L
IN3	L	H	H
IN4	H	L	L
IN5	H	L	H
IN6	H	H	L
IN7	H	H	H

CONVERTER CHARACTERISTICS

The Converter
 The heart of this single chip data acquisition system is its 8-bit analog-to-digital converter. The converter is designed

to give fast, accurate, and repeatable conversions over a wide range of temperatures. The converter is partitioned into 3 major sections: the 256R ladder network, the successive approximation register, and the comparator. The converter's digital outputs are positive true.

The 256R ladder network approach (Figure 1) was chosen over the conventional $R/2R$ ladder because of its inherent monotonicity, which guarantees no missing digital codes. Monotonicity is particularly important in closed loop feedback control systems. A non-monotonic relationship can cause oscillations that will be catastrophic for the system. Additionally, the 256R network does not cause load variations on the reference voltage.

The bottom resistor and the top resistor of the ladder network in Figure 1 are not the same value as the remainder of the network. The difference in these resistors causes the output characteristic to be symmetrical with the zero and full-scale points of the transfer curve. The first output transition occurs when the analog signal has reached +1/2 LSB and succeeding output transitions occur every 1 LSB later up to full-scale.

The successive approximation register (SAR) performs 8 iterations to approximate the input voltage. For any SAR type converter, n-iterations are required for an n-bit converter. Figure 2 shows a typical example of a 3-bit converter. In the ADC0808, ADC0809, the approximation technique is extended to 8 bits using the 256R network.

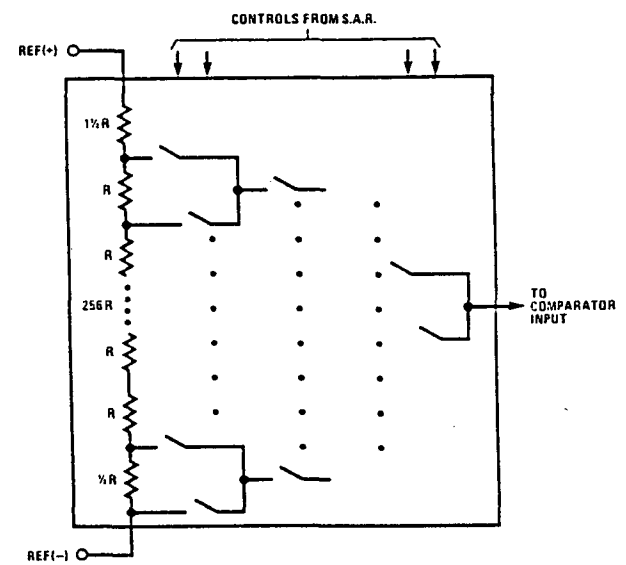


FIGURE 1. Resistor Ladder and Switch Tree

Functional Description (Continued)

The A/D converter's successive approximation register (SAR) is reset on the positive edge of the start conversion (SC) pulse. The conversion is begun on the falling edge of the start conversion pulse. A conversion in process will be interrupted by receipt of a new start conversion pulse. Continuous conversion may be accomplished by tying the end-of-conversion (EOC) output to the SC input. If used in this mode, an external start conversion pulse should be applied after power up. End-of-conversion will go low between 0 and 8 clock pulses after the rising edge of start conversion.

The most important section of the A/D converter is the comparator. It is this section which is responsible for the ultimate accuracy of the entire converter. It is also the

comparator drift which has the greatest influence on the repeatability of the device. A chopper-stabilized comparator provides the most effective method of satisfying all the converter requirements.

The chopper-stabilized comparator converts the DC input signal into an AC signal. This signal is then fed through a high gain AC amplifier and has the DC level restored. This technique limits the drift component of the amplifier since the drift is a DC component which is not passed by the AC amplifier. This makes the entire A/D converter extremely insensitive to temperature, long term drift and input offset errors.

Figure 4 shows a typical error curve for the ADC0808 as measured using the procedures outlined in AN-179.

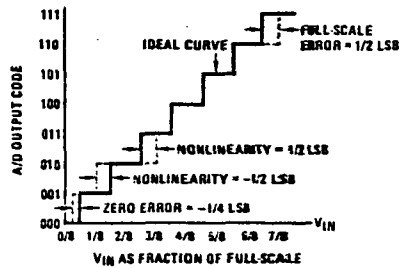


FIGURE 2. 3-Bit A/D Transfer Curve

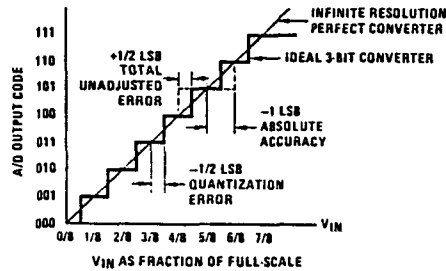


FIGURE 3. 3-Bit A/D Absolute Accuracy Curve

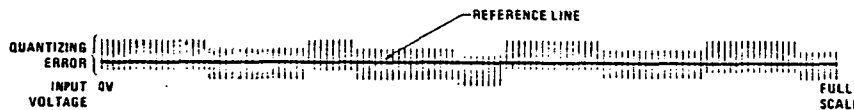
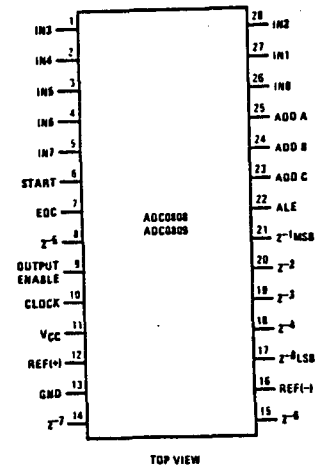


FIGURE 4. Typical Error Curve

Connection Diagram

Dual-In-Line Package



Timing Diagram

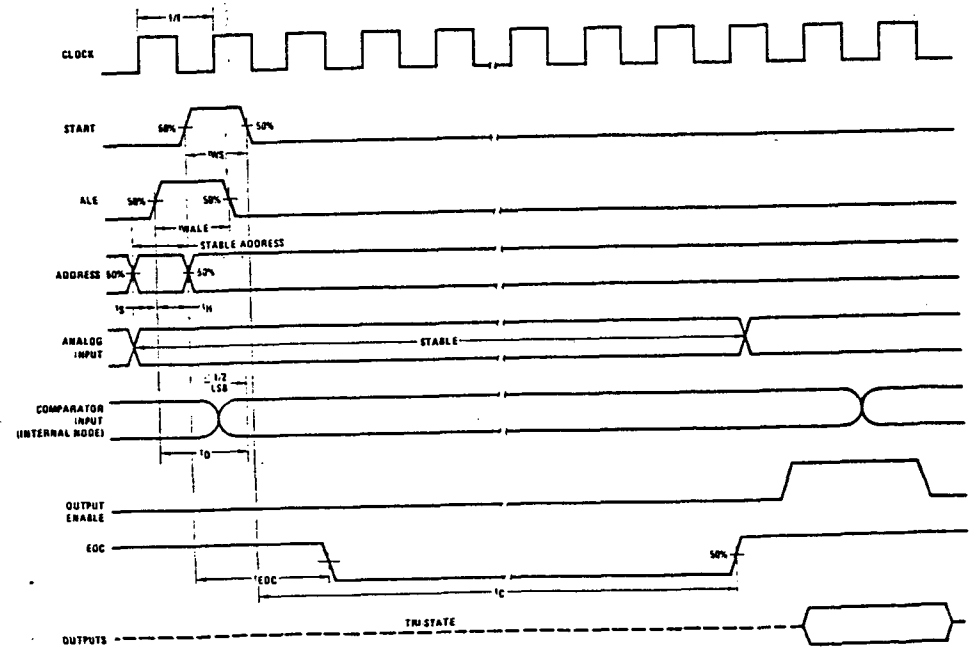


FIGURE 5

Typical Performance Characteristics

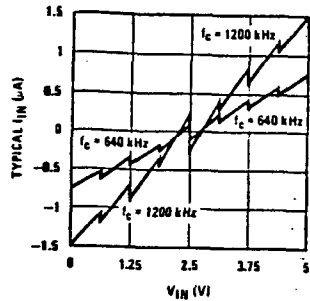


FIGURE 6. Comparator I_{IN} vs V_{IN}
($V_{CC} = V_{REF} = 5V$)

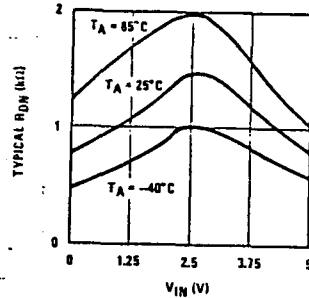


FIGURE 7. Multiplexer R_{ON} vs V_{IN}
($V_{CC} = V_{REF} = 5V$)

TRI-STATE® Test Circuits and Timing Diagrams

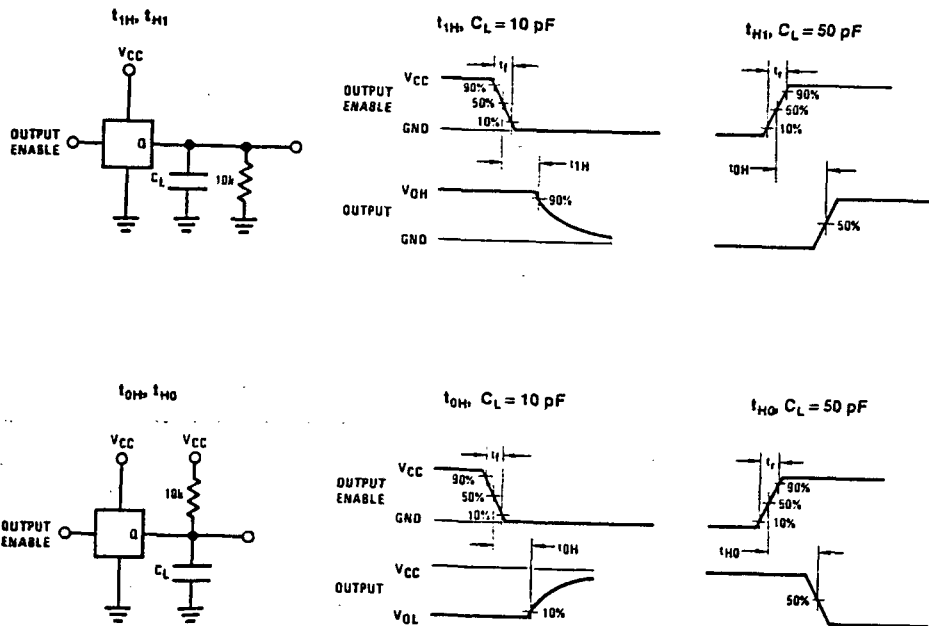


FIGURE 8

Applications Information

OPERATION

1.0 Ratiometric Conversion

The ADC0808, ADC0809 is designed as a complete Data Acquisition System (DAS) for ratiometric conversion systems. In ratiometric systems, the physical variable being measured is expressed as a percentage of full-scale which is not necessarily related to an absolute standard. The voltage input to the ADC0808 is expressed by the equation

$$\frac{V_{IN}}{V_{IS} - V_Z} = \frac{D_X}{D_{MAX} - D_{MIN}} \quad (1)$$

V_{IN} = Input voltage into the ADC0808

V_{IS} = Full-scale voltage

V_Z = Zero voltage

D_X = Data point being measured

D_{MAX} = Maximum data limit

D_{MIN} = Minimum data limit

A good example of a ratiometric transducer is a potentiometer used as a position sensor. The position of the wiper is directly proportional to the output voltage which is a ratio of the full-scale voltage across it. Since the data is represented as a proportion of full-scale, reference requirements are greatly reduced, eliminating a large source of error and cost for many applications. A major advantage of the ADC0808, ADC0809 is that the input voltage range is equal to the supply range so the transducers can be connected directly across the supply and their outputs connected directly into the multiplexer inputs, (Figure 9).

Ratiometric transducers such as potentiometers, strain gauges, thermistor bridges, pressure transducers, etc., are suitable for measuring proportional relationships; however, many types of measurements must be referred to an absolute standard such as voltage or current. This means a system reference must be used which relates the full-scale voltage to the standard volt. For example, if $V_{CC} = V_{REF} = 5.12V$, then the full-scale range is divided into 256 standard steps. The smallest standard step is 1 LSB which is then 20 mV.

2.0 Resistor Ladder Limitations

The voltages from the resistor ladder are compared to the selected input 8 times in a conversion. These voltages are coupled to the comparator via an analog switch tree which is referenced to the supply. The voltages at the top, center and bottom of the ladder, must be controlled to maintain proper operation.

The top of the ladder, Ref(+), should not be more positive than the supply, and the bottom of the ladder, Ref(-), should not be more negative than ground. The center of the ladder voltage must also be near the center of the supply because the analog switch tree changes from N-channel switches to P-channel switches. These limitations are automatically satisfied in ratiometric systems and can be easily met in ground referenced systems.

Figure 10 shows a ground referenced system with a separate supply and reference. In this system, the supply must be trimmed to match the reference voltage. For instance, if a 5.12V is used, the supply should be adjusted to the same voltage within 0.1V.

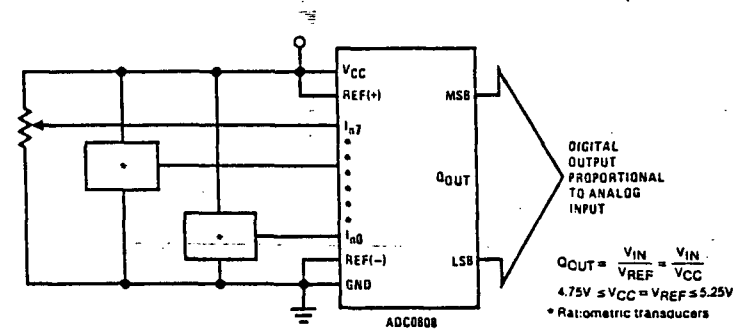


FIGURE 9. Ratiometric Conversion System

$$Q_{OUT} = \frac{V_{IN}}{V_{REF}} = \frac{V_{IN}}{V_{CC}}$$

4.75V ≤ V_{CC} = V_{REF} ≤ 5.25V
• Ratiometric transducers

Applications Information (Continued)

The ADC0808 needs less than a millamp of supply current so developing the supply from the reference is readily accomplished. In Figure 11 a ground referenced system is shown which generates the supply from the reference. The buffer shown can be an op amp of sufficient drive to supply the millamp of supply current and the desired bus drive, or if a capacitive bus is driven by the outputs a large capacitor will supply the transient supply current as seen in Figure 12. The LM301 is overcompensated to insure stability when loaded by the 10 μ F output capacitor.

The top and bottom ladder voltages cannot exceed V_{CC} and ground, respectively, but they can be symmetrically less than V_{CC} and greater than ground. The center of the ladder voltage should always be near the center of the supply. The sensitivity of the converter can be increased, (i.e., size of the LSB steps decreased) by using a symmetrical reference system. In Figure 13, a 2.5V reference is symmetrically centered about $V_{CC}/2$ since the same current flows in identical resistors. This system with a 2.5V reference allows the LSB bit to be half the size of a 5V reference system.

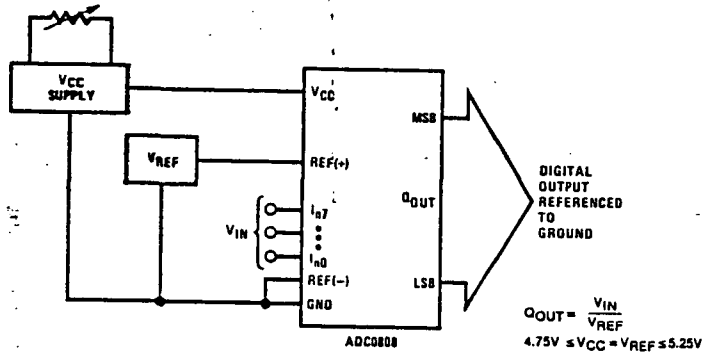


FIGURE 10. Ground Referenced Conversion System Using Trimmed Supply

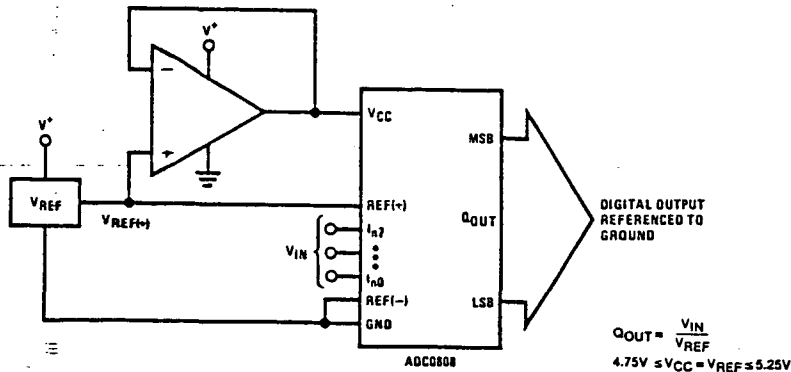


FIGURE 11. Ground Referenced Conversion System with Reference Generating V_{CC} Supply

Applications Information (Continued)

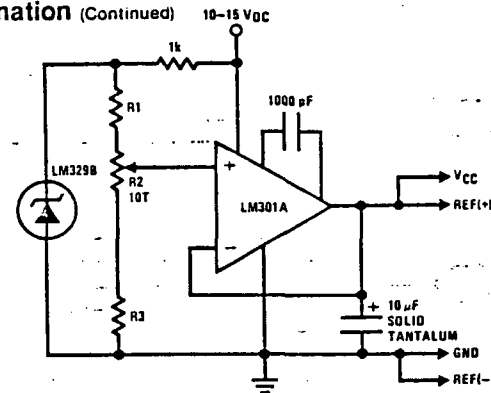


FIGURE 12. Typical Reference and Supply Circuit

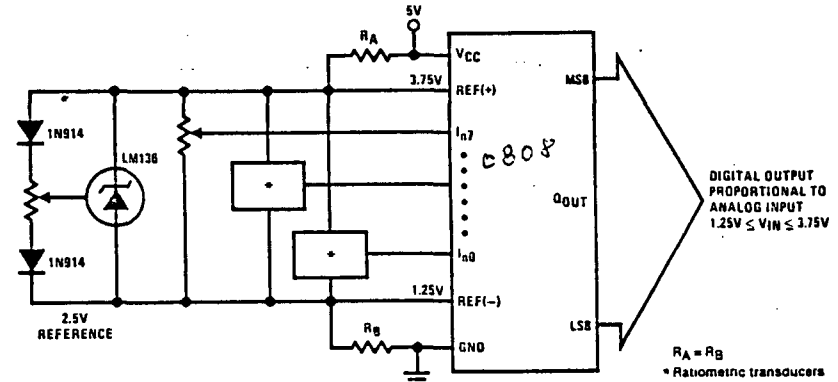


FIGURE 13. Symmetrically Centered Reference

3.0 Converter Equations

The transition between adjacent codes N and $N + 1$ is given by:

$$V_{IN} = \left\{ (V_{REF(+)} - V_{REF(-)}) \left[\frac{N}{256} + \frac{1}{512} \right] \pm V_{TUE} \right\} + V_{REF(-)} \quad (2)$$

The center of an output code N is given by:

$$V_{IN} = \left\{ (V_{REF(+)} - V_{REF(-)}) \left[\frac{N}{256} \right] \pm V_{TUE} \right\} + V_{REF(-)} \quad (3)$$

The output code N for an arbitrary input are the integers within the range:

$$N = \frac{V_{IN} - V_{REF(-)}}{V_{REF(+)} - V_{REF(-)}} \times 256 \pm \text{Absolute Accuracy} \quad (4)$$

where: V_{IN} = Voltage at comparator input

$V_{REF(+)}$ = Voltage at Ref(+)

$V_{REF(-)}$ = Voltage at Ref(-)

V_{TUE} = Total unadjusted error voltage (typically

$V_{REF(+)} \div 512$)

4.0 Analog Comparator Inputs

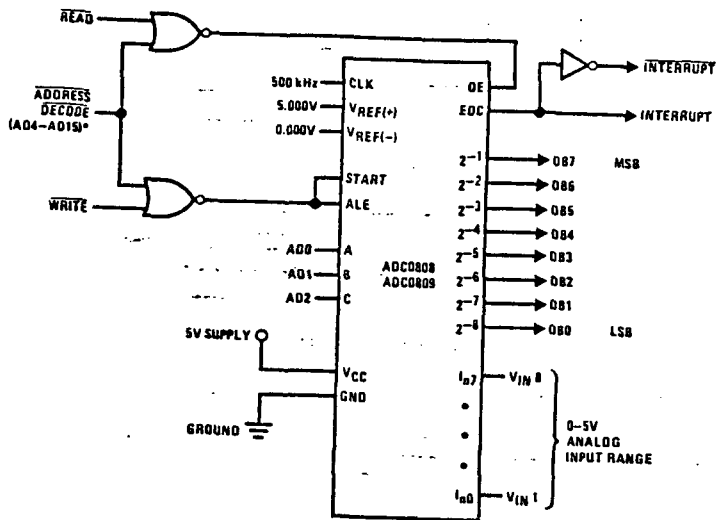
The dynamic comparator input current is caused by the periodic switching of on-chip stray capacitances. These are connected alternately to the output of the resistor ladder/switch tree network and to the comparator input as part of the operation of the chopper stabilized comparator.

The average value of the comparator input current varies directly with clock frequency and with V_{IN} as shown in Figure 6.

If no filter capacitors are used at the analog inputs and the signal source impedances are low, the comparator input current should not introduce converter errors, as the transient created by the capacitance discharge will die out before the comparator output is strobed.

If input filter capacitors are desired for noise reduction and signal conditioning they will tend to average out the dynamic comparator input current. It will then take on the characteristics of a DC bias current whose effect can be predicted conventionally.

Typical Application



* Address latches needed for 6085 and SC/MP interfacing the ADC0808 to a microprocessor

MICROPROCESSOR INTERFACE TABLE

PROCESSOR	READ	WRITE	INTERRUPT (COMMENT)
8080	MEMR	MEMW	INTR (Thru RST Circuit)
8085	RD	WR	INTR (Thru RST Circuit)
Z-80	RD	WR	INT (Thru RST Circuit, Mode 0)
SC/MP	NRDS	NWDS	SA (Thru Sense A)
6800	VMA \times 2 R/W	VMA \times 2 R/W	IRQA or IRQB (Thru PIA)

Ordering Information

TEMPERATURE RANGE		-40°C to +85°C		-55°C to +125°C
Error	$\pm 1/2$ Bit Unadjusted	ADC0808CCN	ADC0808CCJ	ADC0808CJ
	± 1 Bit Unadjusted	ADC0809CCN		
Package Outline		N28A Molded DIP	J28A Hermetic DIP	J28A Hermetic DIP

National Semiconductor

A to D, D to A

ADC0816, ADC0817

ADC0816, ADC0817 8-Bit μ P Compatible A/D Converters with 16-Channel Multiplexer

General Description

The ADC0816, ADC0817 data acquisition component is a monolithic CMOS device with an 8-bit analog-to-digital converter, 16-channel multiplexer and microprocessor compatible control logic. The 8-bit A/D converter uses successive approximation as the conversion technique. The converter features a high impedance chopper stabilized comparator, a 256R voltage divider with analog switch tree and a successive approximation register. The 16-channel multiplexer can directly access any one of 16 single-ended analog signals, and provides the logic for additional channel expansion. Signal conditioning of any analog input signal is eased by direct access to the multiplexer output, and to the input of the 8-bit A/D converter.

The device eliminates the need for external zero and full-scale adjustments. Easy interfacing to microprocessors is provided by the latched and decoded multiplexer address inputs and latched TTL TRI-STATE[®] outputs.

The design of the ADC0816, ADC0817 has been optimized by incorporating the most desirable aspects of several A/D conversion techniques. The ADC0816, ADC0817 offers high speed, high accuracy, minimal temperature dependence, excellent long-term accuracy and repeatability, and consumes minimal power. These features make this device ideally suited to applications from process and machine control to consumer and automotive applications. For similar performance in an 8-channel, 28-pin,

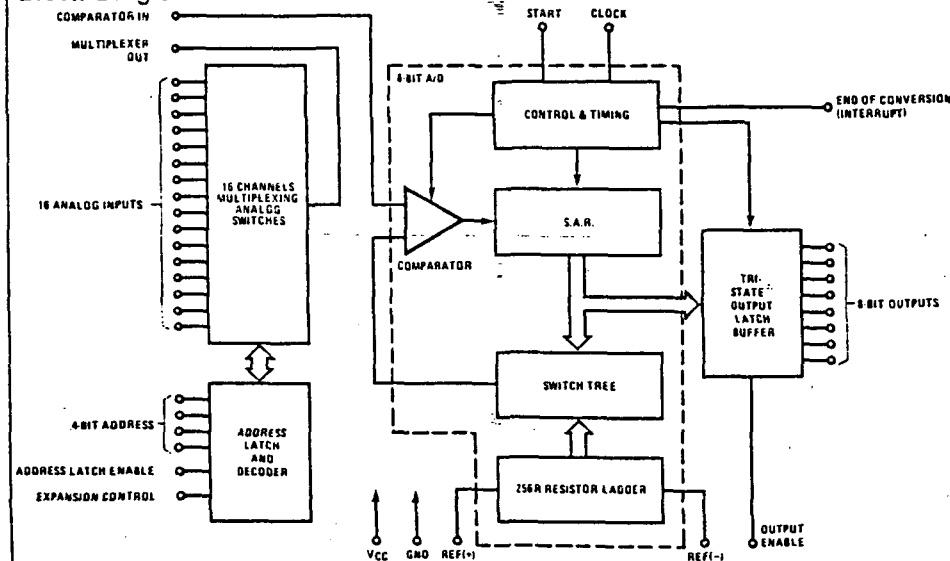
8-bit A/D converter, see the ADC0808, ADC0809 data sheet. (See AN-258 for more information.)

Features

- Resolution — 8-bits
- Total unadjusted error — $\pm 1/2$ LSB and ± 1 LSB
- No missing codes
- Conversion time — 100 μ s
- Single supply — 5 V_{DC}
- Operates ratiometrically or with 5 V_{DC} or analog span adjusted voltage reference
- 16-channel multiplexer with latched control logic
- Easy interface to all microprocessors, or operates "stand alone"
- Outputs meet T²L voltage level specifications
- 0V to 5V analog input voltage range with single 5V supply
- No zero or full-scale adjust required
- Standard hermetic or molded 40-pin DIP package
- Temperature range — -40°C to +85°C or -55°C to +125°C
- Low power consumption — 15 mW
- Latched TRI-STATE[®] output
- Direct access to "comparator in" and "multiplexer out" for signal conditioning

TRI-STATE[®] is a registered trademark of National Semiconductor Corp.

Block Diagram



Typical Electrical Characteristics

TIL 113
TIL 119

Electrical Characteristics—Output Transistor (Darlington) $T_A = 25^\circ\text{C}$

Symbol	Characteristic	Min	Typ	Max	Units	Test Conditions
V_{CEO}	Collector-to-Emitter Breakdown Voltage	30			V	$I_C = 1.0\text{ mA}$, $I_F = 0$
V_{CBO}	Collector-to-Base Breakdown Voltage TIL 113	30			V	$I_C = 10\ \mu\text{A}$, $I_F = 0$
V_{ECO}	Emitter-to-Collector Breakdown Voltage TIL 119	7.0			V	$I_E = 10\ \mu\text{A}$, $I_F = 0$
V_{EBO}	Emitter-to-Base Breakdown Voltage TIL 113	7.0			V	$I_E = 10\ \mu\text{A}$, $I_F = 0$
I_{CEO}	Collector-to-Emitter Leakage Current			100	nA	$V_{CE} = 10\text{ V}$, $I_F = 0$
β_{FE}	Forward Current Gain TIL 113		15 k			$V_{CE} = 1.0\text{ V}$, $I_C = 10\text{ mA}$, $I_F = \phi$

Electrical Characteristics—Coupled $T_A = 25^\circ\text{C}$

Symbol	Characteristic	Min	Typ	Max	Units	Test Conditions
I_C	Collector Output Current (Pulsed) TIL 113	30	100		mA	$I_F = 10\text{ mA}$, $V_{CE} = 1.0\text{ V}$
	TIL 119	30	160		mA	$I_F = 10\text{ mA}$, $V_{CE} = 2.0\text{ V}$
V_{ISO}	Isolation Voltage (Note 2)	1.5 k			V	Peak $V = 500\text{ V}$
R_{ISO}	Isolation Resistance	10^{11}			Ω	
$V_{CE(sat)}$	Collector-to-Emitter Saturation Voltage TIL 113			1.0	V	$I_C = 125\text{ mA}$, $I_B = 0$
	TIL 119			1.0	V	$I_F = 50\text{ mA}$, $I_C = 10\text{ mA}$, $I_F = 10\text{ mA}$
C_{ISO}	Isolation Capacitance		1.0	1.3	pF	$V = 0$
t_r, t_f	Rise and Fall Time (Note 1) TIL 113		300		μs	$f = 1.0\text{ MHz}$, $I_C = 125\text{ mA}$, $V_{CC} = 15\text{ V}$, $R_L = 100\ \Omega$
t_r, t_f	Rise and Fall Time (Note 1) TIL 119		300		μs	$I_C = 2.5\text{ mA}$, $V_{CC} = 10\text{ V}$, $R_L = 100\ \Omega$

Notes

- Rise time is defined as the time for the (base collector) current to rise from 10% to 90% of peak value. Fall time is defined as the time required for the current to decrease from 90% to 10% of peak value.
- Isolation voltage defined as minimum of 5 s continuous application.

Optically-Coupled Isolator

Optoelectronic Products

4N25, 4N26
4N27, 4N28

General Description

The 4N25, 4N26, 4N27, and 4N28 series of opto-isolators has a silicon npn Planar phototransistor coupled to a GaAs diode. Each is mounted in a 6-pin plastic dual in-line package.

Glassolated™

High Current Transfer Ratio—Typically 50%
500 V to 2500 V Minimum Isolation
Input-To-Output
 $10^{11}\ \Omega$ Isolation Resistance
Low Coupling Capacitance—Typically 1.0 pF

Absolute Maximum Ratings

Maximum Temperature and Humidity
Storage Temperature* -55°C to 150°C
Operating Temperature -55°C to 100°C
Pin Temperature
Soldering, 10 s* 260°C
Total Package Power Dissipation at $T_A = 25^\circ\text{C}$,
LED plus Detector* 250 mW
Derate Linearly from 25°C * 3.3 mW/ $^\circ\text{C}$

Input Diode*

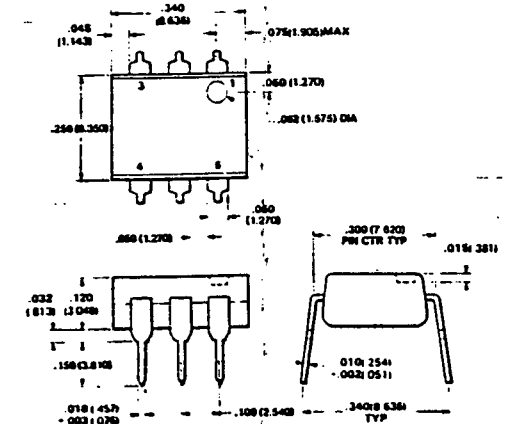
V_R * Reverse Voltage 3.0 V
 I_F * Forward dc Current 80 mA
 I_{pk} * Peak Forward Current, 1 μs pulse width, 300 pps 3.0 A
 P_D * Power Dissipation at $T_A = 25^\circ\text{C}$ 150 mW
Derate Linearly from 25°C 2.0 mW/ $^\circ\text{C}$

Output Transistor

V_{CE} * Collector-to-Emitter Voltage 30 V
 V_{CB} * Collector-to-Base Voltage 70 V
 V_{EC} * Emitter-to-Collector Voltage 7.0 V
 P_D * Power Dissipation at $T_A = 25^\circ\text{C}$ 150 mW
Derate Linearly from 25°C 2.0 mW/ $^\circ\text{C}$

*Indicates JEDEC registered values.

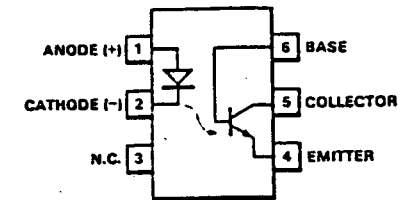
Package Outline



Notes

All dimensions in inches bold and millimeters (parentheses)
Tolerance unless specified = ± 0.015 (± 0.381)

Connection Diagram DIP (Top View)



Pin 1 Anode (+)
Pin 2 Cathode (-) } Input Diode
Pin 3 N.C.
Pin 4 Emitter
Pin 5 Collector } Output npn
Pin 6 Base } Phototransistor

Typical Electrical Characteristics

4N25, 4N26
4N27, 4N28

Electrical Characteristics—Input Diode $T_A = 25^\circ\text{C}$

Symbol	Characteristic	Min	Typ	Max	Units	Test Conditions
	Forward Voltage		1.2	1.5	V	$I_F = 50\text{ mA}$
	Reverse Leakage Current		0.05	100	μA	$V_R = 3.0\text{ V}$, $R_L = 1\text{ M}\Omega$
	Capacitance		150		pF	$V_R = 0\text{ V}$, $f = 1\text{ MHz}$

Electrical Characteristics—Output Transistor $T_A = 25^\circ\text{C}$

Symbol	Characteristic	Min	Typ	Max	Units	Test Conditions
V_{CE}^*	Collector-to-Emitter Voltage	30	65		V	$I_C = 1.0\text{ mA}$, $I_B = 0$
V_{CB}^*	Collector-to-Base Voltage	70	165		V	$I_C = 100\text{ }\mu\text{A}$, $I_E = 0$
V_{EC}^*	Emitter-to-Collector Voltage	7.0	14		V	$I_E = 100\text{ }\mu\text{A}$, $I_B = 0$
I_{EO}^*	Collector-to-Emitter Leakage Current 4N25, 4N26, 4N27		3.5	50	nA	$V_{CE} = 10\text{ V}$, Base Open
	4N28			100	nA	$V_{CE} = 10\text{ V}$, Base Open
I_{BO}^*	Collector-to-Base Leakage Current		0.1	20	nA	$V_{CB} = 10\text{ V}$, Emitter Open
β_{FE}	Forward Current Gain		250			$V_{CE} = 5.0\text{ V}$, $I_C = 500\text{ }\mu\text{A}$
C_{cb}	Collector-to-Base Capacitance		65		pF	$V_{CB} = 0\text{ V}$, $f = 1\text{ MHz}$

Electrical Characteristics—Coupled $T_A = 25^\circ\text{C}$

Symbol	Characteristic	Min	Typ	Max	Units	Test Conditions
V_{IO}^*	Input-to-Output Voltage 4N25	2500			V_{pk}	
	4N26, 4N27	1500			V_{pk}	
	4N28	500			V_{pk}	
$V_{CE(sat)}^*$	Collector-to-Emitter Saturation Voltage		0.2	0.5	V	$I_C = 2.0\text{ mA}$, $I_F = 50\text{ mA}$
I_C^*	Collector Output Current 4N25, 4N28	2.0	5.0		mA	$V_{CE} = 10\text{ V}$, $I_F = 10\text{ mA}$, $I_B = 0$
	4N27, 4N28	1.0	3.0		mA	$V_{CE} = 10\text{ V}$, $I_C = 2.0\text{ mA}$, $R_L = 100\text{ }\Omega$
R_{IO}	Input-to-Output Resistance		10^{11}		Ω	$V_{IO} = 0\text{ V}$, $f = 1.0\text{ MHz}$
$3W$	Collector Bandwidth		300		kHz	
C_{IO}	Input-to-Output Capacitance		1.3		pF	

*Indicates JEDEC registered values.

Optically-Coupled Darlington Isolator

Optoelectronics Products

4N29, 4N30
4N31, 4N32
4N33

General Description

The 4N29, 4N30, 4N31, 4N32 and 4N33 series of optoisolators has a silicon npn-Planar photo-Darlington transistor coupled to a GaAs diode. Each is mounted in a 6-pin plastic DIP package.

High Current Transfer Ratio
1500 V or 2500 V Minimum Isolation
Input-To-Output
 $10^{11}\text{ }\Omega$ Isolation Resistance
Low Coupling Capacitance

Absolute Maximum Ratings

Maximum Temperature and Humidity

Storage Temperature*	-55°C to 150°C
Operating Temperature	-55°C to 100°C
Pin Temperature	280°C
Soldering, 10 s*	280°C
Total Package Power Dissipation at $T_A = 25^\circ\text{C}$	250 mW
(LED plus Detector)*	3.3 mW/ $^\circ\text{C}$
Derate Linearly from 25°C *	

Input Diode*

I_F	Forward dc Current	80 mA
	Continuous*	3.0 V
V_R	Reverse Voltage	3.0 V
I_{pk}	Peak Forward Current (1 μs pulse width, 300 pps)	3.0 A
P_D	Power Dissipation at $T_A = 25^\circ\text{C}$	150 mW
	Derate Linearly from 25°C	2.0 mW/ $^\circ\text{C}$

Output Transistor (Darlington)*

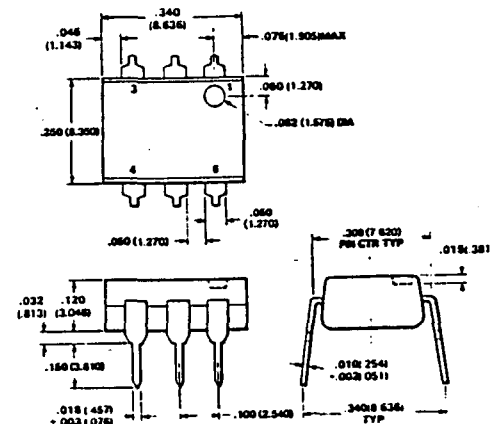
V_{CE}	Collector-to-Emitter Voltage	30 V
V_{CB}	Collector-to-Base Voltage	50 V
V_{EB}	Emitter-to-Base Voltage	8.0 V
V_{EC}	Emitter-to-Collector Voltage	5.0 V
P_D	Power Dissipation at $T_A = 25^\circ\text{C}$	150 mW
	Derate Linearly from 25°C	2.0 mW/ $^\circ\text{C}$

*Indicates JEDEC Registered Data

Electrical Characteristics—Input Diode $T_A = 25^\circ\text{C}$

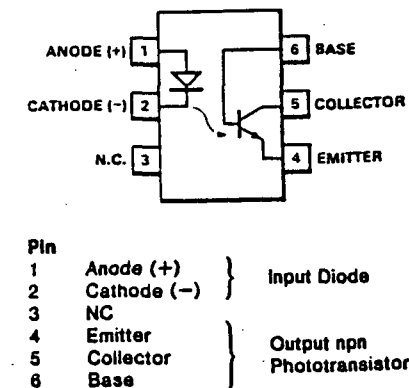
Symbol	Characteristic	Min	Typ	Max	Units	Test Conditions
V_F^*	Forward Voltage		1.2	1.5	V	$I = 50\text{ mA}$
I_{R}^*	Reverse Leakage Current		0.05	100	μA	$V_R = 3.0\text{ V}$
C_{PF}^*	Capacitance		150		pF	$V_R = 0\text{ V}$, $f = 1\text{ MHz}$

Package Outline



Notes
All dimensions in inches bold and millimeters (parentheses)
Tolerance unless specified = ± 0.015 (± 0.381)

Connection Diagram DIP (Top View)



Pin
1 Anode (+)
2 Cathode (-)
3 N.C.
4 Emitter
5 Collector
6 Base

Input Diode
Output npn Phototransistor

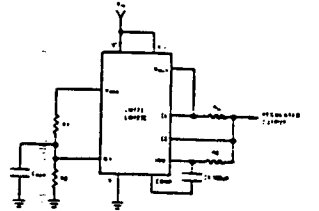
TABLE I RESISTOR VALUES (kΩ) FOR STANDARD OUTPUT VOLTAGE

POSITIVE OUTPUT VOLTAGE	APPLICABLE FIGURES	FIXED OUTPUT ±5%		OUTPUT ADJUSTABLE ±10% (Note 5)			NEGATIVE OUTPUT VOLTAGE	APPLICABLE FIGURES	FIXED OUTPUT ±5%		5% OUTPUT ADJUSTABLE ±10%		
		R1	R2	R1	P1	R2			R1	R2	R1	P1	R2
-3.0	1, 5, 6, 9, 12 (4)	4.12	3.01	1.8	0.5	1.2	-100	7	3.57	102	2.2	10	91
-3.6	1, 5, 6, 9, 12 (4)	3.57	3.65	1.5	0.5	1.5	+250	7	2.57	255	2.2	10	240
-5.0	1, 5, 6, 9, 12 (4)	2.15	4.99	.75	0.5	2.2	-6 (Note 6)	3, (10)	3.57	2.43	1.2	0.5	75
-6.0	1, 5, 6, 9, 12 (4)	1.15	9.04	0.5	0.5	2.7	-9	3, 10	3.48	5.36	1.2	0.5	2.0
-9.0	2, 4, 15, 6, 9, 12	1.87	7.15	.75	1.0	2.7	-12	3, 10	3.57	8.45	1.2	0.5	3.3
-12	2, 4, 15, 6, 9, 12	4.87	7.15	2.0	1.0	3.0	-15	3, 10	3.65	11.5	1.2	0.5	4.3
+15	2, 4, 15, 6, 9, 12	7.87	7.15	3.3	1.0	3.0	-28	3, 10	3.57	24.3	1.2	0.5	10
+28	2, 4, 15, 6, 9, 12	21.0	7.15	5.6	1.0	3.0	-45	8	3.57	41.2	2.2	10	33
+45	7	3.57	48.7	2.2	10	39	-100	8	3.57	97.6	2.2	10	91
+75	7	3.57	78.7	2.2	10	68	-250	8	3.57	249	2.2	10	240

TABLE II FORMULAE FOR INTERMEDIATE OUTPUT VOLTAGES

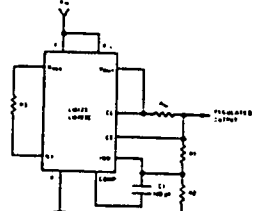
<p>Outputs from +2 to +7 volts (Figures 1, 5, 6, 9, 12, 14)</p> $V_{OUT} = \left[V_{REF} \times \frac{R2}{R1 + R2} \right]$	<p>Outputs from +6 to +250 volts (Figure 7)</p> $V_{OUT} = \left[\frac{V_{SENSE}}{2} \times \frac{R2 - R1}{R1} \right] \cdot R3 \cdot R4$	<p>Current Limiting</p> $I_{LIMIT} = \frac{V_{SENSE}}{R_{SC}}$
<p>Outputs from +7 to +37 volts (Figures 2, 4, 15, 6, 9, 12)</p> $V_{OUT} = \left[V_{REF} \times \frac{R1 - R2}{R2} \right]$	<p>Outputs from -6 to -250 volts (Figures 3, 8, 10)</p> $V_{OUT} = \left[\frac{V_{SENSE}}{2} \times \frac{R1 - R2}{R1} \right] \cdot R3 \cdot R4$	<p>Foldback Current Limiting</p> $I_{SENSE} = \left[\frac{V_{OUT} \cdot R3}{R_{SC} \cdot R4} + \frac{V_{SENSE} \cdot (R3 - R4)}{R_{SC} \cdot R4} \right]$ $I_{SHORTCUT} = \left[\frac{V_{SENSE}}{R_{SC}} \times \frac{R3 - R4}{R4} \right]$

Typical Applications



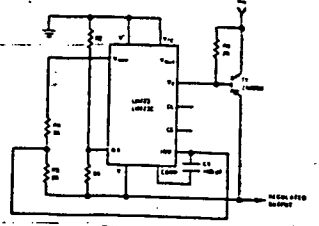
TYPICAL PERFORMANCE
 Regulated Output Voltage: 5V
 Line Regulation (I_{VO} = 3V): 2.5 mV
 Load Regulation (I_L = 50 mA): 1.5 mV

FIGURE 1. Basic Low Voltage Regulator (V_{OUT} = 2 to 7 Volts)



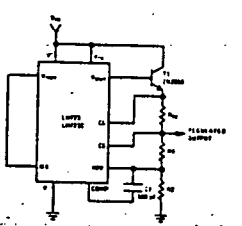
TYPICAL PERFORMANCE
 Regulated Output Voltage: 15V
 Line Regulation (I_{VO} = 3V): 1.5 mV
 Load Regulation (I_L = 50 mA): 4.5 mV

FIGURE 2. Basic High Voltage Regulator (V_{OUT} = 7 to 37 Volts)



TYPICAL PERFORMANCE
 Regulated Output Voltage: -5V
 Line Regulation (I_{VO} = 3V): 1 mV
 Load Regulation (I_L = 100 mA): 2 mV

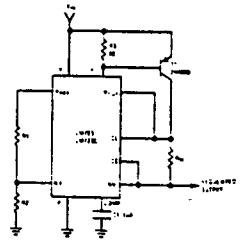
FIGURE 3. Negative Voltage Regulator



TYPICAL PERFORMANCE
 Regulated Output Voltage: +15V
 Line Regulation (I_{VO} = 3V): 1.5 mV
 Load Regulation (I_L = 1A): 15 mV

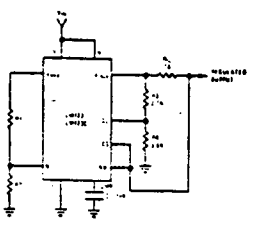
FIGURE 4. Positive Voltage Regulator (External NPN Pass Transistor)

Typical Applications (Continued)



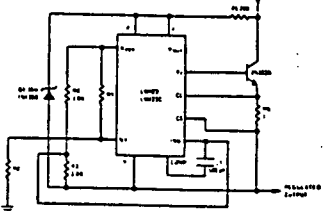
TYPICAL PERFORMANCE
 Regulated Output Voltage: +1V
 Line Regulation (I_{VO} = 3V): 0.5 mV
 Load Regulation (I_L = 1A): 5 mV

FIGURE 5. Positive Voltage Regulator (External PNP Pass Transistor)



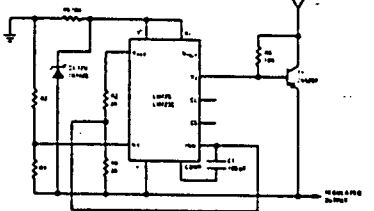
TYPICAL PERFORMANCE
 Regulated Output Voltage: -1V
 Line Regulation (I_{VO} = 2V): 0.5 mV
 Load Regulation (I_L = 10 mA): 1 mV
 Short Circuit Current: 28 mA

FIGURE 6. Foldback Current Limiting



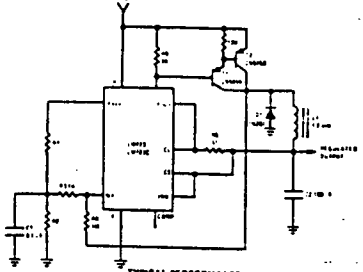
TYPICAL PERFORMANCE
 Regulated Output Voltage: +10V
 Line Regulation (I_{VO} = 20V): 15 mV
 Load Regulation (I_L = 50 mA): 28 mV

FIGURE 7. Positive Floating Regulator



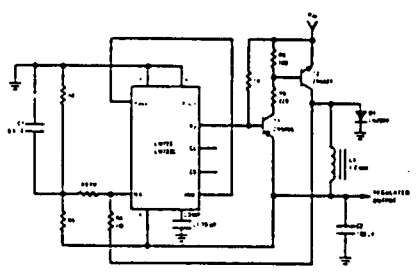
TYPICAL PERFORMANCE
 Regulated Output Voltage: -100V
 Line Regulation (I_{VO} = 20V): 28 mV
 Load Regulation (I_L = 100 mA): 28 mV

FIGURE 8. Negative Floating Regulator



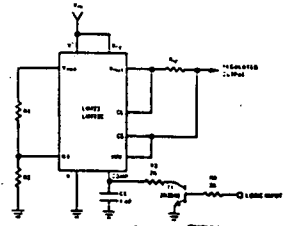
TYPICAL PERFORMANCE
 Regulated Output Voltage: +1V
 Line Regulation (I_{VO} = 20V): 10 mV
 Load Regulation (I_L = 2A): 30 mV

FIGURE 9. Positive Switching Regulator



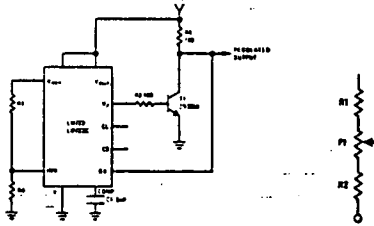
TYPICAL PERFORMANCE
 Regulated Output Voltage: -15V
 Line Regulation (I_{VO} = 20V): 8 mV
 Load Regulation (I_L = 2A): 6 mV

FIGURE 10. Negative Switching Regulator



TYPICAL PERFORMANCE
 Regulated Output Voltage: +5V
 Line Regulation (I_{VO} = 3V): 0.5 mV
 Load Regulation (I_L = 50 mA): 1.5 mV

FIGURE 11. Remote Shutdown Regulator with Current Limiting



TYPICAL PERFORMANCE
 Regulated Output Voltage: +10V
 Line Regulation (I_{VO} = 10V): 0.5 mV
 Load Regulation (I_L = 100 mA): 1.5 mV

FIGURE 12. Shunt Regulator

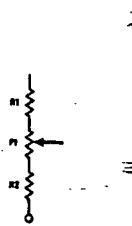
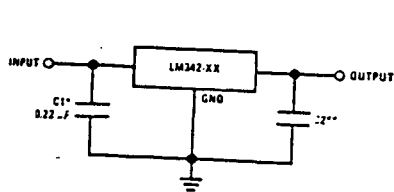


FIGURE 13. Output Voltage Adjust (See Note 5)



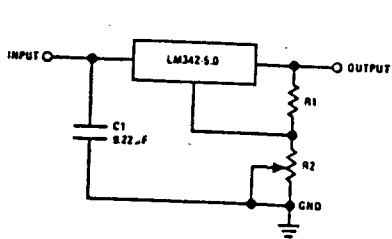
Typical Applications

Fixed Output Regulator



- *Required if the regulator is located far from power supply filter
- **Although not required, C2 does improve transient response. (If needed, use 0.1 μF ceramic disc.)

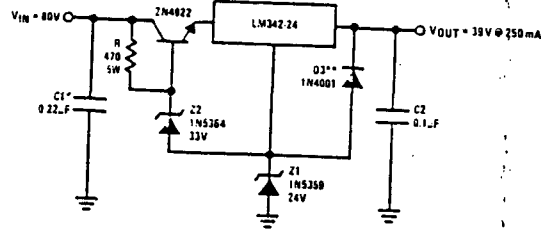
Adjustable Output Regulator



$$V_o = 5V + (5V/R1 + I_Q) R2$$

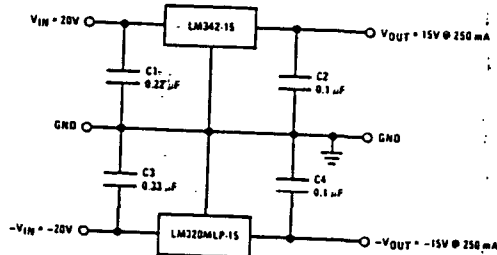
$$5V/R1 > 3I_Q, \text{ Load Regulation (LR)} = \frac{I_Q R2}{(R1 + R2)/R1} \text{ (LR of LM342-05)}$$

High Output Voltage Regulator

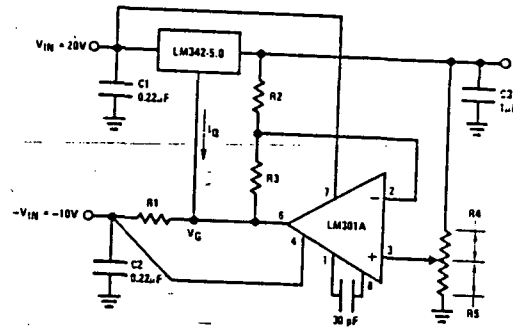


- *Necessary if regulator is located far from the power supply filter
- **D3 aids in full load start-up and protects the regulator during short circuits from high input to output voltage differentials

±15V, 250 mA Dual Power Supply



Variable Output Regulator 0.5V - 18V



- $V_{OUT} = V_G + 5V, R1 = (-V_{IN}/I_Q \text{ LM342})$
- $V_{OUT} = 5V(R2/R4)$ for $(R2 + R3) = (R4 + R5)$
- A 0.5V output will correspond to $(R2/R4) = 0.1, (R3/R4) = 0.9$
- *Solid tantalum

LM723/LM723C Voltage Regulator

General Description

The LM723/LM723C is a voltage regulator designed primarily for series regulator applications. By itself, it will supply output currents up to 150 mA; but external transistors can be added to provide any desired load current. The circuit features extremely low standby current drain, and provision is made for either linear or foldback current limiting. Important characteristics are:

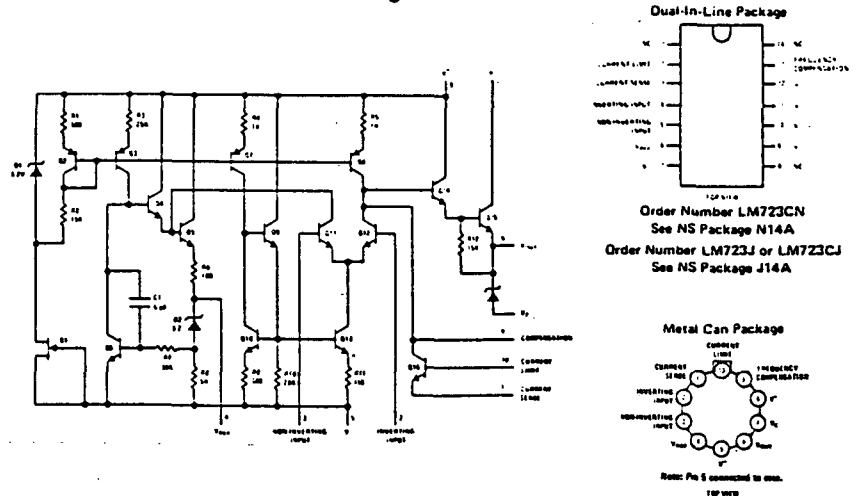
- 150 mA output current without external pass transistor
- Output currents in excess of 10A possible by adding external transistors

- Input voltage 40V max
- Output voltage adjustable from 2V to 37V
- Can be used as either a linear or a switching regulator.

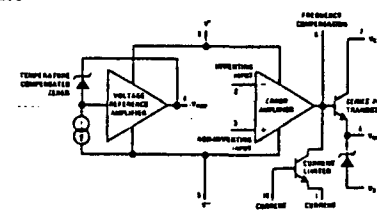
The LM723/LM723C is also useful in a wide range of other applications such as a shunt regulator, a current regulator or a temperature controller.

The LM723C is identical to the LM723 except that the LM723C has its performance guaranteed over a 0°C to 70°C temperature range, instead of -55°C to +125°C.

Schematic and Connection Diagrams*



Equivalent Circuit*



*Pin numbers refer to metal can package.

Controllers provide a solution for Asynchronous, byte Synchronous (IBM Bisync) and Bit Synchronous (HDLC/SDLC) protocols support. It is optimized for high-speed applications requiring the flexibility of the protocol support and the integration of multiple communications channels.

The 82530 SCC (Serial Communications Controller) is another dual channel multiprotocol controller. It contains new functions including on-chip baud rate generators, digital phase locked loops, various data encoding/decoding schemes and extensive diagnostic capabilities. All these added features reduce the need for external logic and greatly improve the reliability and maintainability of the system.

Distributed Intelligence Systems

The 8044/8744 is a microcontroller with an on-chip serial communication processor. It simplifies control of remote subsystems (subsystems that are physically separated from the host CPU and communicate over a serial link).

The 8044 and 8051 CPUs are identical. The serial communication is handled by an additional processor called the Serial Interface Unit (SIU). The SIU operates concurrently with the CPU and offers a high level of intelligence and performance for HDLC/SDLC based communications. The SIU can handle 2.4 Mbps in Half-Duplex mode.

In addition to controlling communications with the host CPU, the 8044 provides significant peripheral control. Examples include local keyboard, CRT and printer control as well as design of network for Distributed Intelligence Systems (Medical instrumentation, CATV, PABX, etc. . .)

Detailed 8044/8744 information is contained in the Intel Microcontroller Handbook.

Instrumentation

The 8291 A, 8292, and 8293 family of components provide complete, high-performance support for IEEE-488 (GPIB) standard interface. GPIB is used in instrumentation applications.

The 8291A implements the Talker/Listener functions of the GPIB.

The 8292 provides the controller functions. Operating in tandem with the 8291A, it complements its interface functions to provide a full-capability GPIB interface.

The 8293 is a low-power, high-current, HMOS 8-line transceiver. It provides the electrical interface to the GPIB.

Local Area Networks

Intel has developed the first complete VLSI solution for Local Area Networks (LANs) and Ethernet in particular: the 82586 Local Area Network Coprocessor and the 82501 ESI (Ethernet Serial Interface).

Four on-chip DMA channels allow the 82586 to operate as a bus master. The 82586 manages the entire process of transmitting and receiving frames, thereby relieving the host processor of the tasks of managing the communication interface to the network.

An extensive set of diagnostic capabilities, implemented in silicon, simplifies the design of more reliable local networks and facilitates their maintenance. In order to take full advantage of the LAN concept and CSMA/CD access method, the 82586 architecture is software configurable. This allows the 82586 to be "customized" for other applications including serial backplanes (serial peripheral interconnection), low cost short distance LANs, broadband networks and medium speed (1-2 Mbps) LANs.

The 82501 is designed to work directly with the 82586 in Ethernet applications. The major functions of the ESI are to generate the 10 MHz transmit clock for the 82586, to perform Manchester encoding/decoding of transmitted/received frames, and to provide the electrical interface to the Ethernet transceiver cable.

The Intel Data Communications product family provides a wide range of solutions for the needs of data communications systems.

PROGRAMMABLE COMMUNICATION INTERFACE

- Synchronous and Asynchronous Operation
- Synchronous 5-8 Bit Characters; Internal or External Character Synchronization; Automatic Sync Insertion
- Asynchronous 5-8 Bit Characters; Clock Rate—1, 16 or 64 Times Baud Rate; Break Character Generation; 1, 1½, or 2 Stop Bits; False Start Bit Detection; Automatic Break Detect and Handling
- Synchronous Baud Rate—DC to 64K Baud
- Asynchronous Baud Rate—DC to 19.2K Baud
- Full-Duplex, Double-Buffered Transmitter and Receiver
- Error Detection—Parity, Overrun and Framing
- Compatible with an Extended Range of Intel Microprocessors
- 28-Pin DIP Package
- All Inputs and Outputs are TTL Compatible
- Available in EXPRESS
 - Standard Temperature Range
 - Extended Temperature Range

The Intel® 8251A is the enhanced version of the industry standard, Intel 8251 Universal Synchronous Asynchronous Receiver/Transmitter (USART), designed for data communications with Intel's microprocessor families such as MCS-48, 80, 85, and iAPX-86, 88. The 8251A is used as a peripheral device and is programmed by the CPU to operate using virtually any serial data transmission technique presently in use (including IBM "bi-sync"). The USART accepts data characters from the CPU in parallel format and then converts them into a continuous serial data stream for transmission. Simultaneously, it can receive serial data streams and convert them into parallel data characters for the CPU. The USART will signal the CPU whenever it can accept a new character for transmission or whenever it has received a character for the CPU. The CPU can read the complete status of the USART at any time. These include data transmission errors and control signals such as SYNDET, TxEMPTY. The chip is fabricated using N-channel silicon gate technology.

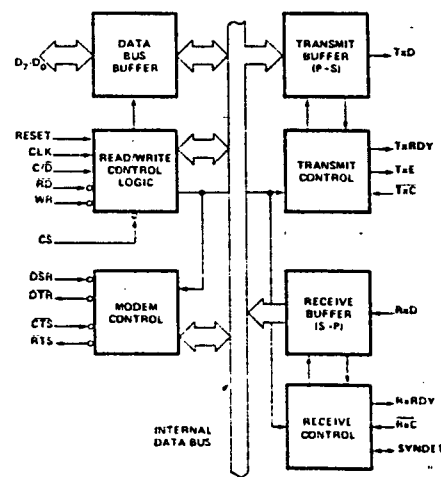


Figure 1: Block Diagram

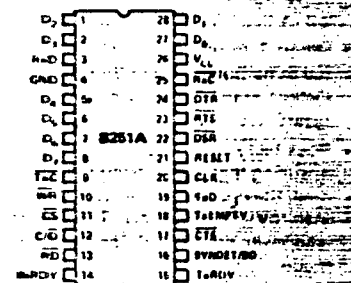


Figure 2: Pin Configuration

FEATURES AND ENHANCEMENTS

8251A is an advanced design of the industry standard USART, the Intel® 8251. The 8251A starts with an extended range of Intel microprocessors and maintains compatibility with 8251. Familiarization time is minimal because of compatibility and involves only knowing the additional features and enhancements, and reviewing AC and DC specifications of the 8251A.

8251A incorporates all the key features of the 8251 and has the following additional features and enhancements:

Double-buffered data paths with separate registers for control, status, Data In, and Data Out, which considerably simplifies control programming and minimizes CPU overhead.

In asynchronous operations, the Receiver detects and handles "break" automatically, relieving the CPU of this task.

Refined Rx initialization prevents the Receiver from starting when in "break" state, preventing unwanted interrupts from a disconnected USART.

At the conclusion of a transmission, Tx D line will always return to the marking state unless SBRK is programmed.

Tx Enable logic enhancement prevents a Tx Disable command from halting transmission until all data previously written has been transmitted. The logic also prevents the transmitter from turning off in the middle of a word.

When External Sync Detect is programmed, Internal Sync Detect is disabled, and an External Sync Detect status is provided via a flip-flop which clears itself upon a status read.

Possibility of false sync detect is minimized by ensuring that if double character sync is programmed, the characters be contiguously detected and also by clearing the Rx register to all ones whenever Enter Hunt command is issued in Sync mode.

As long as the 8251A is not selected, the RD and WR do not affect the internal operation of the device.

The 8251A Status can be read at any time but the status update will be inhibited during status read.

The 8251A is free from extraneous glitches and has enhanced AC and DC characteristics, providing higher speed and better operating margins.

Synchronous Baud rate from DC to 64K.

FUNCTIONAL DESCRIPTION

General

The 8251A is a Universal Synchronous/Asynchronous Receiver/Transmitter designed for a wide range of Intel microcomputers such as 8048, 8080, 8085, 8086 and 8088. Like other I/O devices in a microcomputer system, its functional configuration is programmed by the system's software for maximum flexibility. The 8251A can support most serial data techniques in use, including IBM "bi-sync."

In a communication environment an interface device must convert parallel format system data into serial format for transmission and convert incoming serial format data into parallel system data for reception. The interface device must also delete or insert bits or characters that are functionally unique to the communication technique. In essence, the interface should appear "transparent" to the CPU, a simple input or output of byte-oriented system data.

Data Bus Buffer

This 3-state, bidirectional, 8-bit buffer is used to interface the 8251A to the system Data Bus. Data is transmitted or received by the buffer upon execution of INput or OUTput instructions of the CPU. Control words, Command words and Status information are also transferred through the Data Bus Buffer. The Command Status, Data-In and Data-Out registers are separate, 8-bit registers communicating with the system bus through the Data Bus Buffer.

This functional block accepts inputs from the system Control bus and generates control signals for overall device operation. It contains the Control Word Register and Command Word Register that store the various control formats for the device functional definition.

RESET (Reset)

A "high" on this input forces the 8251A into an "Idle" mode. The device will remain at "Idle" until a new set of control words is written into the 8251A to program its functional definition. Minimum RESET pulse width is 6 t_{cy} (clock must be running).

A command reset operation also puts the device into the "Idle" state.

CLK (Clock)

The CLK input is used to generate internal device timing and is normally connected to the Phase 2 (TTL) output of the Clock Generator. No external inputs or outputs are referenced to CLK but the frequency of CLK must be greater than 30 times the Receiver or Transmitter data bit rates.

WR (Write)

A "low" on this input informs the 8251A that the CPU is writing data or control words to the 8251A.

RD (Read)

A "low" on this input informs the 8251A that the CPU is reading data or status information from the 8251A.

C/D (Control/Data)

This input, in conjunction with the WR and RD inputs, informs the 8251A that the word on the Data Bus is either a data character, control word or status information.

1 = CONTROL/STATUS; 0 = DATA.

CS (Chip Select)

A "low" on this input selects the 8251A. No reading or writing will occur unless the device is selected. When CS is high, the Data Bus is in the float state and RD and WR have no effect on the chip.

Modem Control

The 8251A has a set of control inputs and outputs that can be used to simplify the interface to almost any modem. The modem control signals are general purpose in nature and can be used for functions other than modem control, if necessary.

DSR (Data Set Ready)

The DSR input signal is a general-purpose, 1-bit inverting input port. Its condition can be tested by the CPU using a Status Read operation. The DSR input is normally used to test modem conditions such as Data Set Ready.

DTR (Data Terminal Ready)

The DTR output signal is a general-purpose, 1-bit inverting output port. It can be set "low" by programming the appropriate bit in the Command Instruction word. The DTR output signal is normally used for modem control such as Data Terminal Ready.

RTS (Request to Send)

The RTS output signal is a general-purpose, 1-bit inverting output port. It can be set "low" by programming the appropriate bit in the Command Instruction word. The RTS output signal is normally used for modem control such as Request to Send.

CTS (Clear to Send)

A "low" on this input enables the 8251A to transmit serial data if the Tx Enable bit in the Command byte is set to a "one." If either a Tx Enable off or CTS off condition occurs while the Tx is in operation, the Tx will transmit all the data in the USART, written prior to Tx Disable command before shutting down.

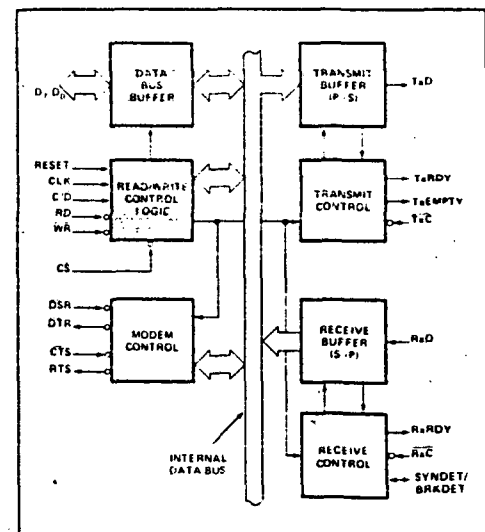


Figure 3. 8251A Block Diagram Showing Data Bus Buffer and Read/Write Logic Functions

C/D	RD	WR	CS	
0	0	1	0	8251A DATA → DATA BUS
0	1	0	0	DATA BUS → 8251A DATA
1	0	1	0	STATUS → DATA BUS
1	1	0	0	DATA BUS → CONTROL
X	1	1	0	DATA BUS = 3-STATE
X	X	X	1	DATA BUS = 3-STATE

Transmitter Buffer

The Transmitter Buffer accepts parallel data from the Data Bus Buffer, converts it to a serial stream, inserts the appropriate characters or bits (based on the communication technique) and outputs a composite serial stream of data on the TxD output pin on the falling edge of Tx̄C. The transmitter will begin transmission upon being enabled if CTS = 0. The Tx̄D line will be held in the marking state immediately upon a master Reset or when Tx̄Enable or CTS is off or the transmitter is empty.

Transmitter Control

Transmitter Control manages all activities associated with the transmission of serial data. It accepts and issues signals both externally and internally to accomplish this function.

TxDY (Transmitter Ready)

This output signals the CPU that the transmitter is ready to accept a data character. The TxDY output pin can be used as an interrupt to the system since it is masked by Tx̄Enable; or, for Polled operation, the CPU can check TxDY using a Status Read operation. TxDY is automatically reset by the leading edge of WR when a data character is loaded from the CPU.

Note that when using the Polled operation, the TxDY status bit is *not* masked by Tx̄Enable but will only indicate the Empty/Full Status of the Tx Data Input Register.

TxE (Transmitter Empty)

When the 8251A has no characters to send the Tx̄EMPTY output will go "high." It resets upon receiving a character from CPU if the transmitter is enabled. Tx̄EMPTY remains high when the transmitter is disabled. Tx̄EMPTY can be used to monitor the end of a transmission mode, so that the CPU "knows" when to "turn the line around" in the full-duplex operational mode.

In the Synchronous mode, a "high" on the output indicates that a character has not been received and the SYNC character or characters are about to be or are being transmitted automatically at "inters." Tx̄EMPTY does not go low when the SYNC characters are being shifted out.

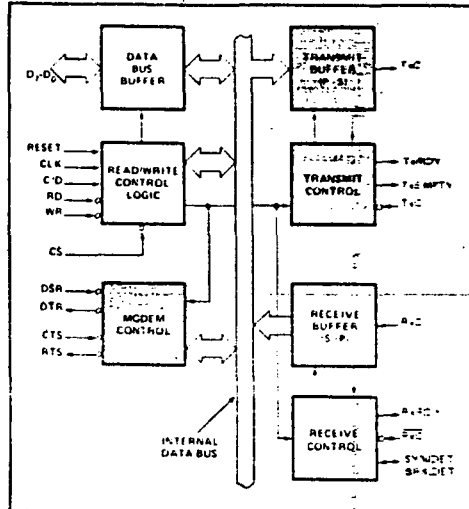


Figure 4. 8251A Block Diagram Showing Modem and Transmitter Buffer and Control Functions

TxC (Transmitter Clock)

The Transmitter Clock controls the rate at which the character is to be transmitted. In the Synchronous transmission mode, the Baud Rate (1x) is equal to the Tx̄C frequency. In Asynchronous transmission mode, the baud rate is a fraction of the actual Tx̄C frequency. A portion of the mode instruction selects this factor; it can be 1, 1/16 or 1/64 the Tx̄C.

For Example:

If Baud Rate equals 110 Baud,
Tx̄C equals 110 kHz in the 1x mode.
Tx̄C equals 1.72 kHz in the 16x mode.
Tx̄C equals 7.04 kHz in the 64x mode.

The falling edge of Tx̄C shifts the serial data out of the 8251A.

Receiver Buffer

The Receiver accepts serial data, converts this serial input to parallel format, checks for bits or characters that are unique to the communication technique and sends an "assembled" character to the CPU. Serial data is input to Rx̄D pin, and is clocked in on the rising edge of Rx̄C.

Receiver Control

This functional block manages all receiver-related activities which consists of the following features.

The Rx̄D initialization circuit prevents the 8251A from mistaking an unused input line for an active low data line in the "break condition." Before starting to receive serial characters on the Rx̄D line, a valid "1" must first be detected after a chip master Reset. Once this has been determined, a search for a valid low (Start bit) is enabled. This feature is only active in the asynchronous mode, and is only done once for each master Reset.

The False Start bit detection circuit prevents false starts due to a transient noise spike by first detecting the falling edge and then strobing the nominal center of the Start bit (Rx̄D = low).

Parity error detection sets the corresponding status bit.

The Framing Error status bit is set if the Stop bit is absent at the end of the data byte (asynchronous mode).

RxDY (Receiver Ready)

This output indicates that the 8251A contains a character that is ready to be input to the CPU. Rx̄RDY can be connected to the interrupt structure of the CPU or, for polled operation, the CPU can check the condition of Rx̄RDY using a Status Read operation.

Rx̄Enable, when off, holds Rx̄RDY in the Reset Condition. For Asynchronous mode, to set Rx̄RDY, the Receiver must be enabled to sense a Start Bit and a complete character must be assembled and transferred to the Data Output Register. For Synchronous mode, to set Rx̄RDY, the Receiver must be enabled and a character must finish assembly and be transferred to the Data Output Register.

Failure to read the received character from the Rx Data Output Register prior to the assembly of the next Rx Data character will set overrun condition error and the previous character will be written over and lost. If the Rx Data is being read by the CPU when the internal transfer is occurring, overrun error will be set and the old character will be lost.

RxC (Receiver Clock)

The Receiver Clock controls the rate at which the character is to be received. In Synchronous Mode, the Baud Rate (1x) is equal to the actual frequency of Rx̄C. In Asynchronous Mode, the Baud Rate is a fraction of the actual Rx̄C frequency. A portion of the mode instruction selects this factor: 1, 1/16 or 1/64 the Rx̄C.

For example:

Baud Rate equals 300 Baud, if
Rx̄C equals 300 Hz in the 1x mode;
Rx̄C equals 4800 Hz in the 16x mode;
Rx̄C equals 19.2 kHz in the 64x mode.

Baud Rate equals 2400 Baud, if
Rx̄C equals 2400 Hz in the 1x mode;
Rx̄C equals 38.4 kHz in the 16x mode;
Rx̄C equals 153.6 kHz in the 64x mode.

Data is sampled into the 8251A on the rising edge of Rx̄C.

NOTE: In most communications systems, the 8251A will be handling both the transmission and reception operations of a single link. Consequently, the Receive and Transmit Baud Rates will be the same. Both Tx̄C and Rx̄C will require identical frequencies for this operation and can be tied together and connected to a single frequency source (Baud Rate Generator) to simplify the interface.

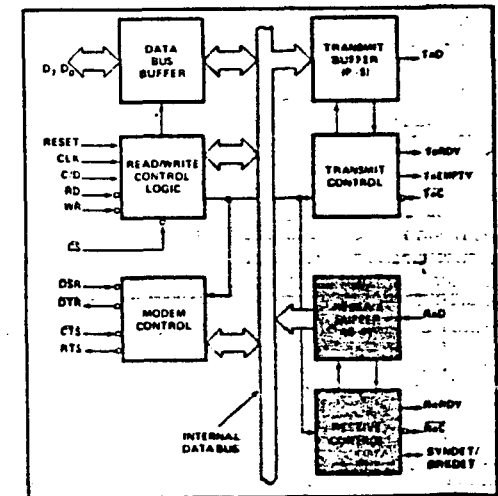


Figure 5. 8251A Block Diagram Showing Receiver Buffer and Control Functions

SYNDET (SYNC Detect/ BRKDET Break Detect)

This pin is used in Synchronous Mode for SYNDET and may be used as either input or output, programmable through the Control Word. It is reset to output mode low upon RESET. When used as an output (internal Sync mode), the SYNDET pin will go "high" to indicate that the 8251A has located the SYNC character in the Receive mode. If the 8251A is programmed to use double Sync characters (bi-sync), then SYNDET will go "high" in the middle of the last bit of the second Sync character. SYNDET is automatically reset upon a Status Read operation.

Used as an input (external SYNC detect mode), a rising going signal will cause the 8251A to start assembling data characters on the rising edge of the next RxC . Once in SYNC, the "high" input signal can be removed. When External SYNC Detect is programmed, Internal SYNC Detect is disabled.

BREAK (Async Mode Only)

This output will go high whenever the receiver remains low through two consecutive stop bit sequences (including the start bits, data bits, and parity bits). Break Detect may also be read as a Status bit. It is reset only upon a master chip Reset or Rx Data returning to a "one" state.

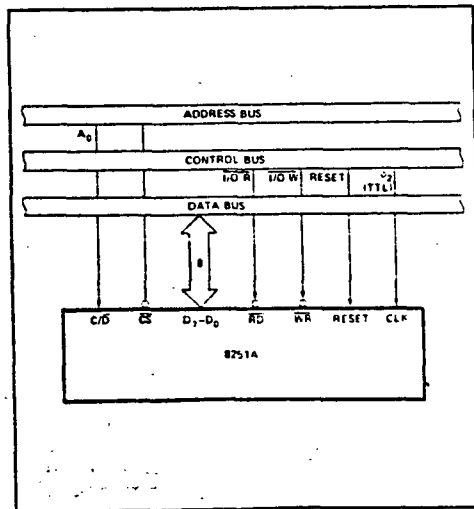


Figure 6. 8251A Interface to 8080 Standard System Bus

DETAILED OPERATION DESCRIPTION

General

The complete functional definition of the 8251A is programmed by the system's software. A set of control words must be sent out by the CPU to initialize the 8251A to support the desired communications format. These control words will program the: BAUD RATE, CHARACTER LENGTH, NUMBER OF STOP BITS, SYNCHRONOUS or ASYNCHRONOUS OPERATION, EVEN/ODD/OFF PARITY, etc. In the Synchronous Mode, options are also provided to select either internal or external character synchronization.

Once programmed, the 8251A is ready to perform its communication functions. The TxRDY output is raised "high" to signal the CPU that the 8251A is ready to receive a data character from the CPU. This output (TxRDY) is reset automatically when the CPU writes a character into the 8251A. On the other hand, the 8251A receives serial data from the MODEM or I/O device. Upon receiving an entire character, the RxRDY output is raised "high" to signal the CPU that the 8251A has a complete character ready for the CPU to fetch. RxRDY is reset automatically upon the CPU data read operation.

The 8251A cannot begin transmission until the Tx Enable (Transmitter Enable) bit is set in the Command Instruction and it has received a Clear To Send (CTS) input. The TxD output will be held in the marking state upon Reset.

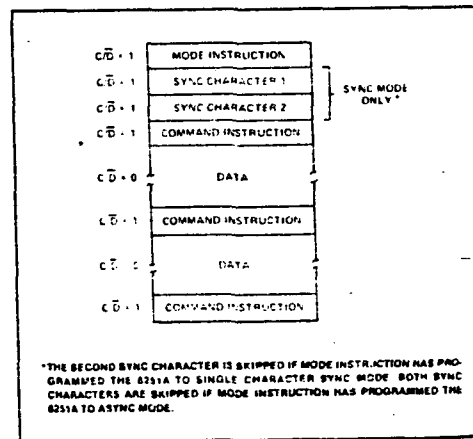


Figure 7. Typical Data Block

Programming the 8251A

Prior to starting data transmission or reception, the 8251A must be loaded with a set of control words generated by the CPU. These control signals define the complete functional definition of the 8251A and must immediately follow a Reset operation (internal or external).

The control words are split into two formats:

1. Mode Instruction
2. Command Instruction

Mode Instruction

This instruction defines the general operational characteristics of the 8251A. It must follow a Reset operation (internal or external). Once the Mode Instruction has been written into the 8251A by the CPU, SYNC characters or Command Instructions may be written.

Command Instruction

This instruction defines a word that is used to control the actual operation of the 8251A.

Both the Mode and Command Instructions must conform to a specified sequence for proper device operation (see Figure 7). The Mode Instruction must be written immediately following a Reset operation, prior to using the 8251A for data communication.

All control words written into the 8251A after the Mode Instruction will load the Command Instruction. Command Instructions can be written into the 8251A at any time in the data block during the operation of the 8251A. To return to the Mode Instruction format, the master Reset bit in the Command Instruction word can be set to initiate an internal Reset operation which automatically places the 8251A back into the Mode Instruction format. Command Instructions must follow the Mode Instructions or Sync characters.

Mode Instruction Definition

The 8251A can be used for either Asynchronous or Synchronous data communication. To understand how the Mode Instruction defines the functional operation of the 8251A, the designer can best view the device as two separate components, one Asynchronous and the other Synchronous, sharing

the same package. The format definition can be changed only after a master chip Reset. For explanation purposes the two formats will be isolated.

NOTE: When parity is enabled it is not considered as one of the data bits for the purpose of programming the word length. The actual parity bit received on the Rx Data line cannot be read on the Data Bus. In the case of a programmed character length of less than 8 bits, the least significant Data Bus bits will hold the data; unused bits are "don't care" when writing data to the 8251A, and will be "zeros" when reading the data from the 8251A.

Asynchronous Mode (Transmission)

Whenever a data character is sent by the CPU the 8251A automatically adds a Start bit (low level) followed by the data bits (least significant bit first), and the programmed number of Stop bits to each character. Also, an even or odd Parity bit is inserted prior to the Stop bit(s), as defined by the Mode Instruction. The character is then transmitted as a serial data stream on the TxD output. The serial data is shifted out on the falling edge of TxC at a rate equal to 1, 1/16, or 1/64 that of the TxC , as defined by the Mode Instruction. BREAK characters can be continuously sent to the TxD if commanded to do so.

When no data characters have been loaded into the 8251A the TxD output remains "high" (marking) unless a Break (continuously low) has been programmed.

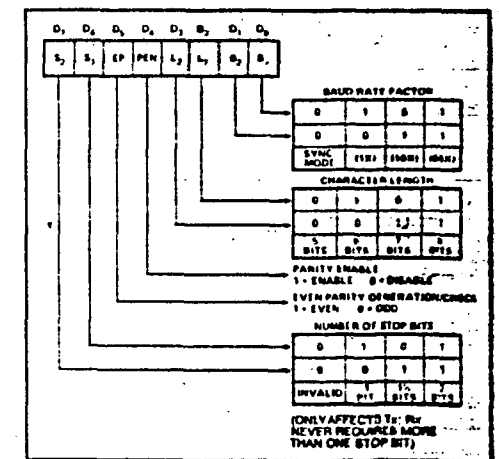


Figure 8. Mode Instruction Format, Asynchronous Mode

Asynchronous Mode (Receive)

The RxD line is normally high. A falling edge on this line triggers the beginning of a START bit. The validity of this START bit is checked by again strobing this bit at its nominal center (16X or 64X mode only). If a low is detected again, it is a valid START bit, and the bit counter will start counting. The bit counter thus locates the center of the data bits, the parity bit (if it exists) and the stop bits. If parity error occurs, the parity error flag is set. Data and parity bits are sampled on the RxD pin with the rising edge of RxC. If a low level is detected as the STOP bit, the Framing Error flag will be set. The STOP bit signals the end of a character. Note that the receiver returns only one stop bit, regardless of the number of stop bits programmed. This character is then loaded into the parallel I/O buffer of the 8251A. The RxDY pin is raised to signal the CPU that a character is ready to be fetched. If a previous character has not been fetched by the CPU, the present character is placed in the I/O buffer, and the OVERRUN Error flag is raised (thus the previous character is lost). All of the error flags can be reset by an Error Reset Instruction. The occurrence of any of these errors will not affect the operation of the 8251A.

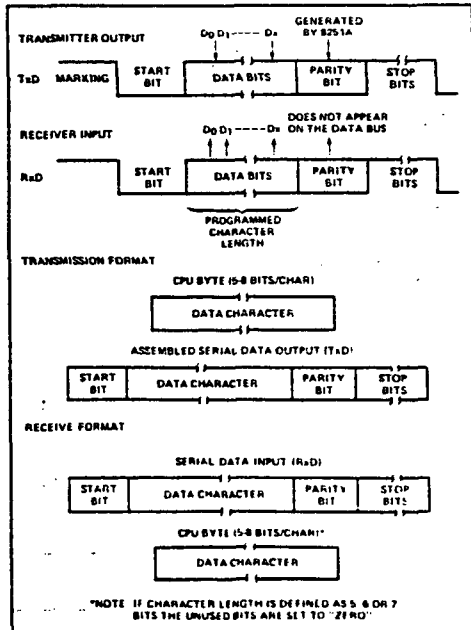
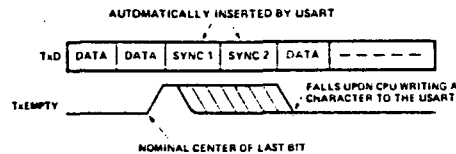


Figure 9. Asynchronous Mode

Synchronous Mode (Transmission)

The TxD output is continuously high until the CPU sends its first character to the 8251A which usually is a SYNC character. When the CTS line goes low, the first character is serially transmitted out. All characters are shifted out on the falling edge of TxC. Data is shifted out at the same rate as the TxC.

Once transmission has started, the data stream at the TxD output must continue at the TxC rate. If the CPU does not provide the 8251A with a data character before the 8251A Transmitter Buffers become empty, the SYNC characters (or character if in single SYNC character mode) will be automatically inserted in the TxD data stream. In this case, the TxEMPTY pin is raised high to signal that the 8251A is empty and SYNC characters are being sent out. TxEMPTY does not go low when the SYNC is being shifted out (see figure below). The TxEMPTY pin is internally reset by a data character being written into the 8251A.



Parity error and overrun error are both checked in the same way as in the Asynchronous Rx mode. Parity is checked when not in Hunt, regardless of whether the Receiver is enabled or not.

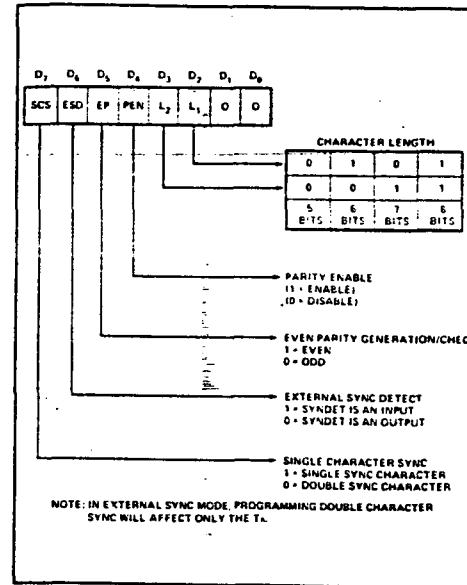


Figure 10. Mode Instruction Format, Synchronous Mode

Synchronous Mode (Receive)

In this mode, character synchronization can be internally or externally achieved. If the SYNC mode has been programmed, ENTER HUNT command should be included in the first command instruction word written. Data on the RxD pin is then sampled on the rising edge of RxC. The content of the Rx buffer is compared at every bit boundary with the first SYNC character until a match occurs. If the 8251A has been programmed for two SYNC characters, the subsequent received character is also compared; when both SYNC characters have been detected, the USART ends the HUNT mode and is in character synchronization. The SYNDT pin is then set high, and is reset automatically by a STATUS READ. If parity is programmed, SYNDT will not be set until the middle of the parity bit instead of the middle of the last data bit.

In the external SYNC mode, synchronization is achieved by applying a high level on the SYNDT pin, thus forcing the 8251A out of the HUNT mode. The high level can be removed after one RxC cycle. An ENTER HUNT command has no effect in the asynchronous mode of operation.

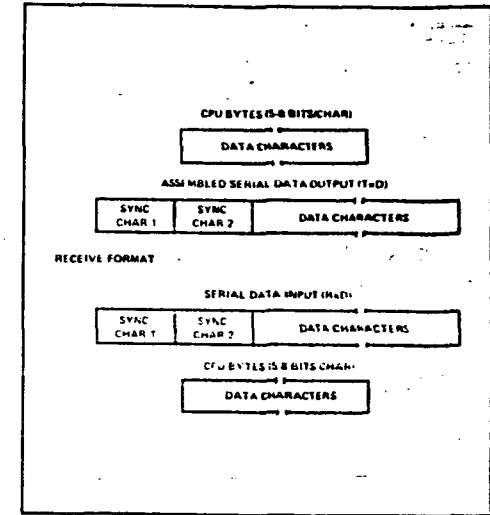


Figure 11. Data Format, Synchronous Mode

COMMAND INSTRUCTION DEFINITION

Once the functional definition of the 8251A has been programmed by the Mode Instruction and the sync characters are loaded (if in Sync Mode) then the device is ready to be used for data communication. The Command Instruction controls the actual operation of the selected format. Functions such as: Enable Transmit/Receive, Error Reset and Modem Controls are provided by the Command Instruction.

Once the Mode Instruction has been written into the 8251A and Sync characters inserted, if necessary, then all further "control writes" (C/D = 1) will load a Command Instruction. A Reset Operation (internal or external) will return the 8251A to the Mode Instruction format.

Note: Internal Reset on Power-up

When power is first applied, the 8251A may come up in the Mode, Sync character or Command format. To guarantee that the device is in the Command Instruction format before the Reset command is issued, it is safest to execute the worst-case initialization sequence (sync mode with two sync characters). Loading three 00Hs consecutively into the device with C/D = 1 configures sync operation and writes two dummy 00H sync characters. An Internal Reset command (40H) may then be issued to return the device to the "Idle" state.

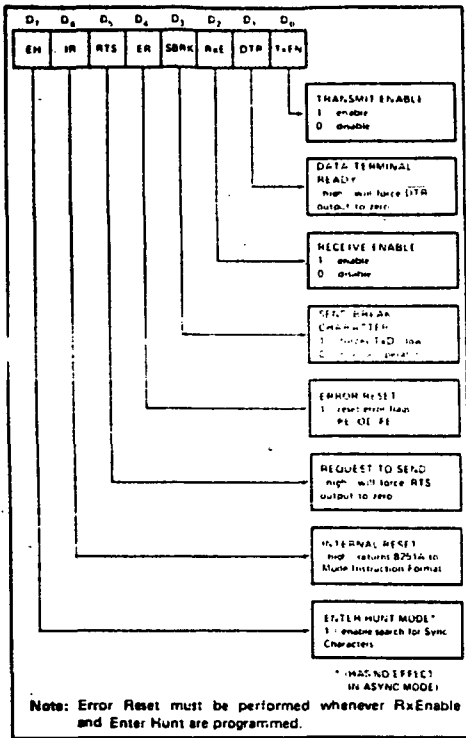


Figure 12. Command Instruction Format

STATUS READ DEFINITION

In data communication systems it is often necessary to examine the "status" of the active device to ascertain if errors have occurred or other conditions that require the processor's attention. The 8251A has facilities that allow the programmer to "read" the status of the device at any time during the functional operation. (Status update is inhibited during status read.)

A normal "read" command is issued by the CPU with $C/\bar{D} = 1$ to accomplish this function.

Some of the bits in the Status Read Format have identical meanings to external output pins so that the 8251A can be used in a completely polled or interrupt-driven environment. TxRDY is an exception.

Note that status update can have a maximum delay of 28 clock periods from the actual event affecting the status.

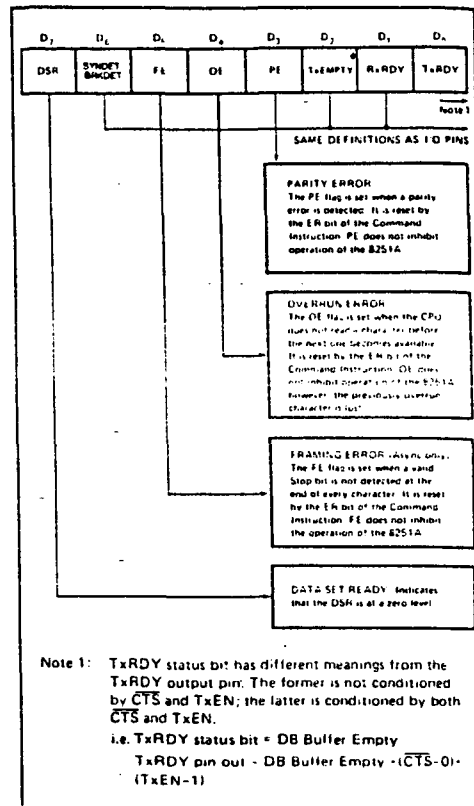


Figure 13. Status Read Format

APPLICATIONS OF THE 8251A

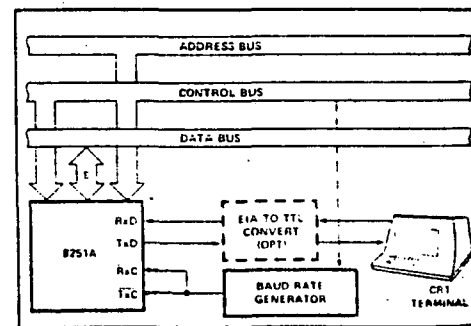


Figure 14. Asynchronous Serial Interface to CRT Terminal, DC—9600 Baud

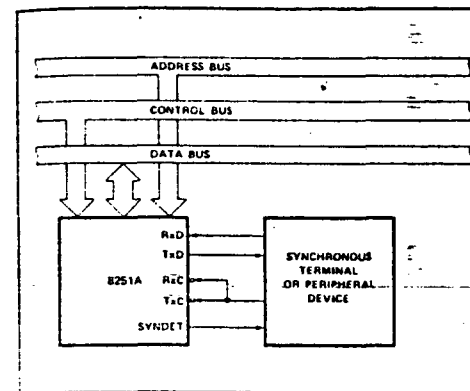


Figure 15. Synchronous Interface to Terminal or Peripheral Device

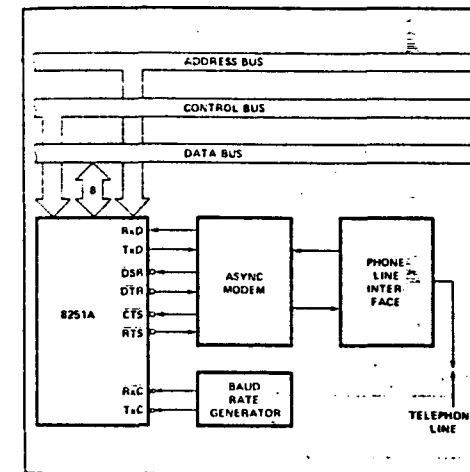


Figure 16. Asynchronous Interface to Telephone Lines

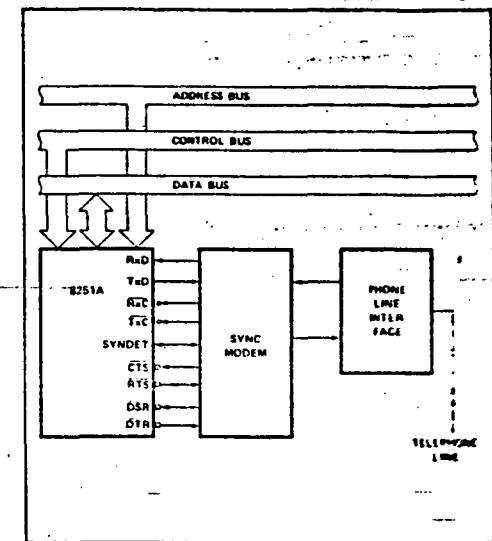


Figure 17. Synchronous Interface to Telephone Lines

ABSOLUTE MAXIMUM RATINGS*

Ambient Temperature Under Bias	0°C to 70°C
Storage Temperature	-65°C to +150°C
Voltage On Any Pin	
With Respect To Ground	-0.5V to +7V
Power Dissipation	1 Watt

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

D.C. CHARACTERISTICS (T_A = 0°C to 70°C, V_{CC} = 5.0V ± 5%, GND = 0V)*

Symbol	Parameter	Min.	Max.	Unit	Test Conditions
V _{IL}	Input Low Voltage	-0.5	0.8	V	
V _{IH}	Input High Voltage	2.0	V _{CC}	V	
V _{OL}	Output Low Voltage		0.45	V	I _{OL} = 2.2 mA
V _{OH}	Output High Voltage	2.4		V	I _{OL} = -400 μA
I _{OFL}	Output Float Leakage		±10	μA	V _{OUT} = V _{CC} TO 0.45V
I _{IL}	Input Leakage		±10	μA	V _{IN} = V _{CC} TO 0.45V
I _{CC}	Power Supply Current		100	mA	All Outputs = High

CAPACITANCE (T_A = 25°C, V_{CC} = GND = 0V)

Symbol	Parameter	Min.	Max.	Unit	Test Conditions
C _{IN}	Input Capacitance		10	pF	f _c = 1MHz
C _{I/O}	I/O Capacitance		20	pF	Unmeasured pins returned to GND

A.C. CHARACTERISTICS (T_A = 0°C to 70°C, V_{CC} = 5.0V ± 10%, GND = 0V)*

Bus Parameters (Note 1)

READ CYCLE

Symbol	Parameter	Min.	Max.	Unit	Test Conditions
t _{AR}	Address Stable Before READ (CS, C _D)	0		ns	Note 2
t _{RA}	Address Hold Time for READ (CS, C _D)	0		ns	Note 2
t _{RR}	READ Pulse Width	250		ns	
t _{RD}	Data Delay from READ		200	ns	3. C _L = 150 pF
t _{DF}	READ to Data Floating	10	100	ns	

WRITE CYCLE

Symbol	Parameter	Min.	Max.	Unit	Test Conditions
t _{AW}	Address Stable Before WRITE	0		ns	
t _{WA}	Address Hold Time for WRITE	0		ns	
t _{WW}	WRITE Pulse Width	250		ns	
t _{DW}	Data Set-Up Time for WRITE	150		ns	
t _{WD}	Data Hold Time for WRITE	20		ns	
t _{RV}	Recovery Time Between WRITES	6		t _{cy}	Note 4

A.C. CHARACTERISTICS (Continued)

OTHER TIMINGS

Symbol	Parameter	Min.	Max.	Unit	Test Conditions
t _{cy}	Clock Period	320	1350	ns	Notes 5, 6
t _g	Clock High Pulse Width	120	t _{cy} - 90	ns	
t _l	Clock Low Pulse Width	90		ns	
t _R , t _F	Clock Rise and Fall Time		20	ns	
t _{DTx}	TxD Delay from Falling Edge of Tx _C		1	μs	
t _{Tx}	Transmitter Input Clock Frequency				
	1x Baud Rate	DC	64	kHz	
	16x Baud Rate	DC	310	kHz	
	64x Baud Rate	DC	615	kHz	
t _{RPW}	Transmitter Input Clock Pulse Width				
	1x Baud Rate	12		t _{cy}	
	16x and 64x Baud Rate	1		t _{cy}	
t _{TPD}	Transmitter Input Clock Pulse Delay				
	1x Baud Rate	15		t _{cy}	
	16x and 64x Baud Rate	3		t _{cy}	
t _{Rx}	Receiver Input Clock Frequency				
	1x Baud Rate	DC	64	kHz	
	16x Baud Rate	DC	310	kHz	
	64x Baud Rate	DC	615	kHz	
t _{RPW}	Receiver Input Clock Pulse Width				
	1x Baud Rate	12		t _{cy}	
	16x and 64x Baud Rate	1		t _{cy}	
t _{RPD}	Receiver Input Clock Pulse Delay				
	1x Baud Rate	15		t _{cy}	
	16x and 64x Baud Rate	3		t _{cy}	
t _{TxRDY}	TxRDY Pin Delay from Center of Last Bit		14	t _{cy}	Note 7
t _{TxRDY CLEAR}	TxRDY ↓ from Leading Edge of WR		400	ns	Note 7
t _{RxRDY}	RxRDY Pin Delay from Center of Last Bit		26	t _{cy}	Note 7
t _{RxRDY CLEAR}	RxRDY ↓ from Leading Edge of RD		400	ns	Note 7
t _{IS}	Internal SYNDET Delay from Rising Edge of Rx _C		26	t _{cy}	Note 7
t _{ES}	External SYNDET Set-Up Time After Rising Edge of Rx _C	18		t _{cy}	Note 7
t _{TxEMPTY}	TxEMPTY Delay from Center of Last Bit		20	t _{cy}	Note 7
t _{WC}	Control Delay from Rising Edge of WRITE (TxEn, DTR, RTS)		8	t _{cy}	Note 7
t _{CR}	Control to READ Set-Up Time (DSR, CTS)	20		t _{cy}	Note 7

*NOTE:

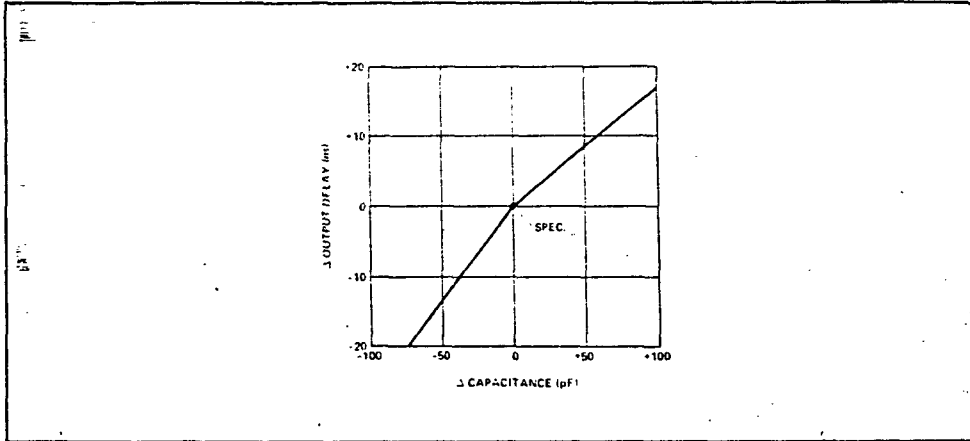
- For Extended Temperature EXPRESS, use MB251A electrical parameters.

A.C. CHARACTERISTICS (Continued)

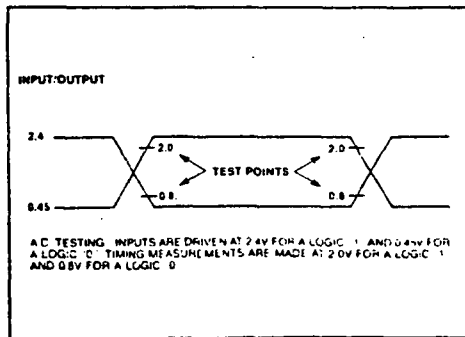
NOTES:

- AC timings measured $V_{OH} = 2.0$, $V_{OL} = 2.0$, $V_{OL} = 0.8$, and with load circuit of Figure 1.
- Chip Select (CS) and Command Data (C/D) are considered as Addresses.
- Assumes that Address is valid before R_D .
- This recovery time is for Mode Initialization only. Write Data is allowed only when $TxRDY = 1$. Recovery Time between Writes for Asynchronous Mode is $8 t_{CY}$ and for Synchronous Mode is $16 t_{CY}$.
- The TxC and Rx C frequencies have the following limitations with respect to CLK: For 1x Baud Rate, f_{Tx} or $f_{Rx} \leq 1/(30 t_{CY})$; For 16x and 64x Baud Rate, f_{Tx} or $f_{Rx} \leq 1/(4.5 t_{CY})$.
- Reset Pulse Width = $6 t_{CY}$ minimum; System Clock must be running during Reset.
- Status update can have a maximum delay of 28 clock periods from the event affecting the status.

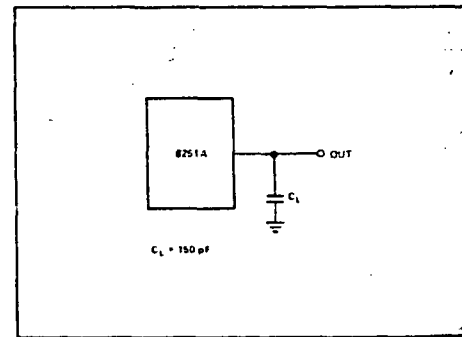
TYPICAL Δ OUTPUT DELAY VS. Δ CAPACITANCE (pF)



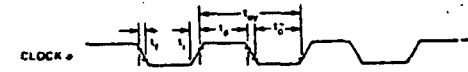
A.C. TESTING INPUT, OUTPUT WAVEFORM



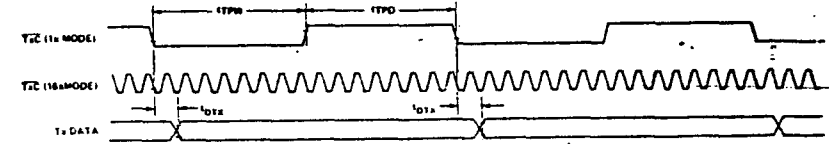
A.C. TESTING LOAD CIRCUIT



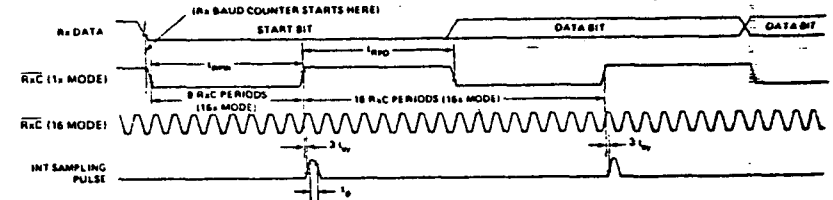
SYSTEM CLOCK INPUT



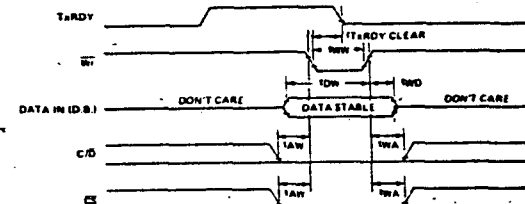
TRANSMITTER CLOCK AND DATA



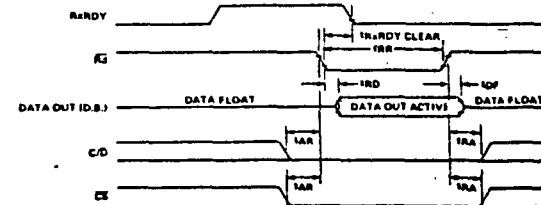
RECEIVER CLOCK AND DATA



WRITE DATA CYCLE (CPU → USART)



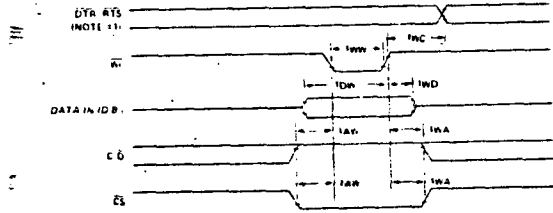
READ DATA CYCLE (CPU ← USART)



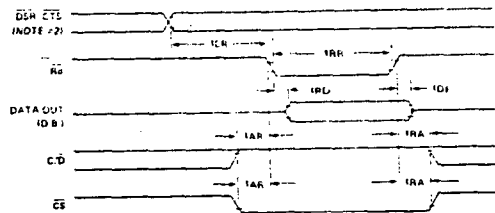
© 1985 Intel Corporation. All rights reserved. Intel and the Intel logo are trademarks of Intel Corporation. 2008

WAVEFORMS (Continued)

WRITE CONTROL OR OUTPUT PORT CYCLE (CPU → USART)

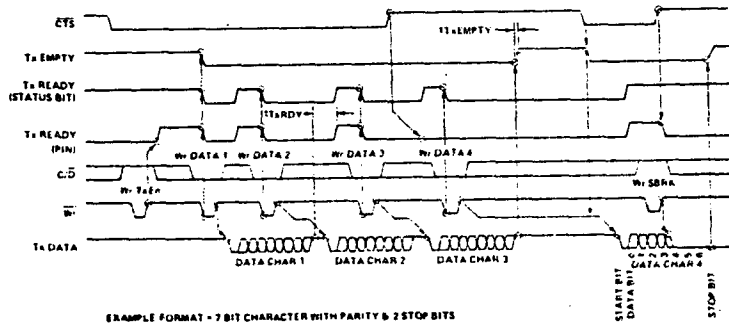


READ CONTROL OR INPUT PORT (CPU ← USART)



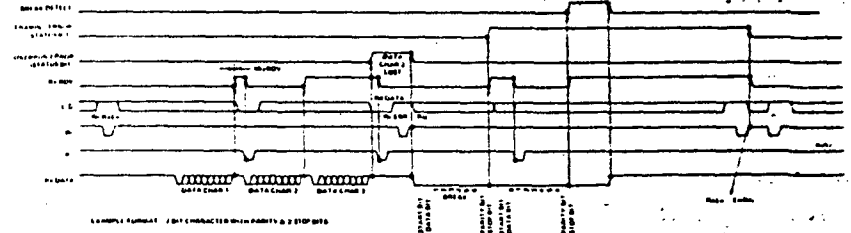
NOTE #1 T_{RC} INCLUDES THE RESPONSE TIMING OF A CONTROL BYTE
NOTE #2 T_{CR} INCLUDES THE EFFECT OF CTS ON THE TXENB CIRCUITRY.

TRANSMITTER CONTROL AND FLAG TIMING (ASYNC MODE)



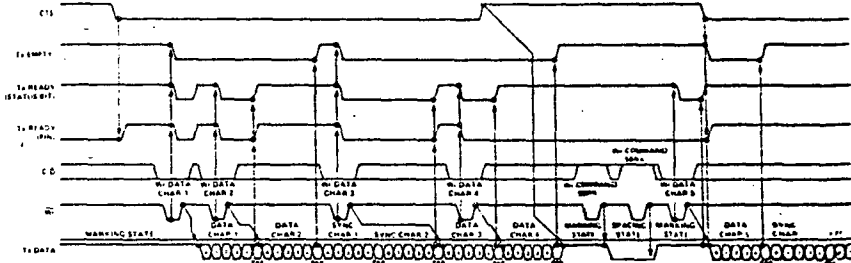
EXAMPLE FORMAT - 7 BIT CHARACTER WITH PARITY & 2 STOP BITS

RECEIVER CONTROL AND FLAG TIMING (ASYNC MODE)



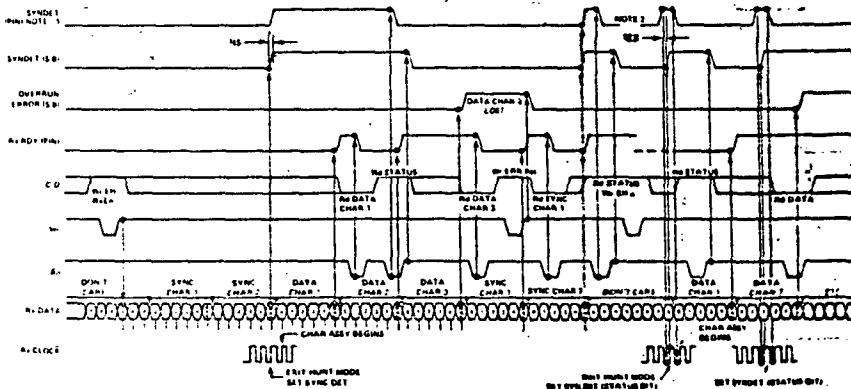
EXAMPLE FORMAT - 7 BIT CHARACTER WITH PARITY & 2 STOP BITS

TRANSMITTER CONTROL AND FLAG TIMING (SYNC MODE)



EXAMPLE FORMAT - 9 BIT CHARACTER WITH PARITY & SYNC CHARACTERS

RECEIVER CONTROL AND FLAG TIMING (SYNC MODE)



NOTE 1 INTERNAL SYNC, 2 SYNC CHARACTERS 8 BITS WITH PARITY
NOTE 2 EXTERNAL SYNC, 2 BITS WITH PARITY



2732A 32K (4K x 8) UV ERASABLE PROM

- 200 ns (2732A-2) Maximum Access Time... HMOS[®]-E Technology
- Compatible with High-Speed 8MHz IAPX 186...Zero WAIT State
- Two Line Control
- Compatible with 12 MHz 8051 Family
- Industry Standard Pinout... JEDEC Approved
- Low Standby Current...30 mA Maximum
- ±10% V_{CC} Tolerance Available
- Intelligent Identifier™ Mode
- TTL Compatible

The Intel 2732A is a 5V only, 32,768 bit ultraviolet erasable and electrically programmable read-only-memory (EPROM). The standard 2732A access time is 250 ns with speed selection (2732A-2) available at 200 ns. The access time is compatible with high performance microprocessors such as the 8 MHz iAPX 186. In these systems, the 2732A allows the microprocessor to operate without the addition of WAIT states.

An important 2732A feature is the separate output control. Output Enable (\overline{OE}), from the Chip Enable control (\overline{CE}). The \overline{OE} control eliminates bus contention in microprocessor systems. Intel's Application Note AP-72 describes the microprocessor system implementation of the \overline{OE} and \overline{CE} controls on Intel's EPROMs. AP-72 is available from Intel's Literature Department.

The 2732A has a standby mode which reduces power consumption without increasing access time. The maximum active current is 125 mA, while the maximum standby current is only 35 mA, a 70% saving. The standby mode is selected by applying the TTL-high signal to the \overline{CE} input.

The 2732A is fabricated with HMOS[®]-E technology, Intel's high-speed N-channel MOS Silicon Gate Technology.

*HMOS is a patented process of Intel Corporation.

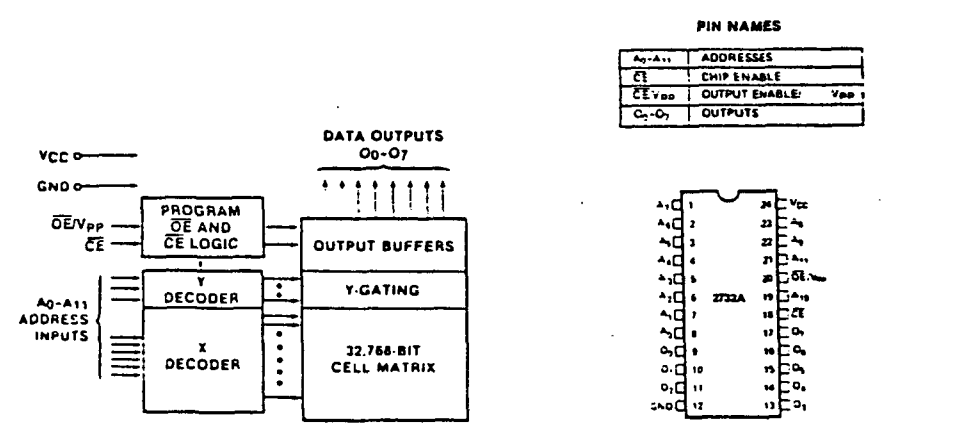


Figure 1. Block Diagram

Figure 2. Pin Configuration

ABSOLUTE MAXIMUM RATINGS*

Temperature Under Bias	-10°C to +80°C
Storage Temperature	-65°C to +125°C
All Input or Output Voltages with Respect to Ground	+6V to -0.3V
Voltage on Pin 22 with Respect to Ground	+13.5V to -0.3V
V _{pp} Supply Voltage with Respect to Ground During Programming	+22V to -0.3V

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

D.C. AND A.C. OPERATING CONDITIONS DURING READ

	2732A/A-2/A-3/A-4	2732A-20/A-25/A-30
Operating Temperature Range	0°C-70°C	0°C-70°C
V _{CC} Power Supply ^{1,2}	5V ± 5%	5V ± 10%

READ OPERATION

D.C. CHARACTERISTICS

Symbol	Parameter	Limits			Units	Conditions
		Min.	Typ. ⁽¹⁾	Max.		
I _{IL}	Input Load Current			10	μA	V _{IN} = 5.5V
I _{LO}	Output Leakage Current			10	μA	V _{OUT} = 5.5V
I _{CC1} ²	V _{CC} Current (Standby)			35	mA	$\overline{CE} = V_{IH}, \overline{OE} = V_{IL}$
I _{CC2} ²	V _{CC} Current (Active)			100	mA	$\overline{OE} = \overline{CE} = V_{IL}$
V _{IL}	Input Low Voltage	-0.1		0.8	V	
V _{IH}	Input High Voltage	2.0		V _{CC} + 1	V	
V _{OL}	Output Low Voltage			0.45	V	I _{OL} = 2.1 mA
V _{OH}	Output High Voltage	2.4			V	I _{OH} = -400 μA
V _{PP} ²	V _{PP} Read Voltage	3.8		V _{CC}	V	V _{CC} = 5.0V ± 0.25V

A.C. CHARACTERISTICS

Symbol	Parameter	2732A-2 2732A-20		2732A 2732A-25		2732A-3 2732A-30		2732A-4		Units	Test Conditions ³
		Min.	Max.	Min.	Max.	Min.	Max.	Min.	Max.		
t _{ACC}	Address to Output Delay		200		250		300		450	ns	$\overline{CE} = \overline{OE} = V_{IL}$
t _{CE}	\overline{CE} to Output Delay		200		250		300		450	ns	$\overline{OE} = V_{IL}$
t _{OE}	\overline{OE} to Output Delay		70		100		150		150	ns	$\overline{CE} = V_{IL}$
t _{DR} ⁽⁴⁾	\overline{OE} High to Output Not Driven	0	60	0	60	0	130	0	130	ns	$\overline{CE} = V_{IL}$
t _{OH}	Output Hold from Addresses, \overline{CE} or \overline{OE} Whichever Occurred First	0		0		0		0		ns	$\overline{CE} = \overline{OE} = V_{IL}$

†A.C. TEST CONDITIONS

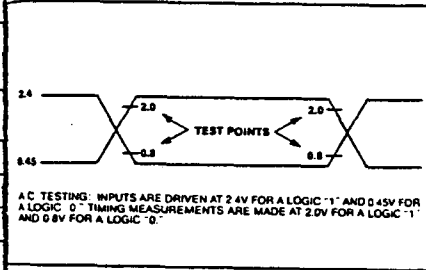
Output Load 1 TTL gate and C_L = 100 pF
 Input Rise and Fall Times ≤ 20 ns
 Input Pulse Levels 0.45V to 2.4V
 Timing Measurement Reference Level:
 Inputs 0.8 and 2.0 V
 Outputs 0.8 and 2.0 V

- NOTES: 1. V_{CC} must be applied simultaneously or before V_{PP} and removed simultaneously or after V_{PP}.
 2. V_{PP} may be connected directly to V_{CC} except during programming. The supply current would then be the sum of I_{CC1} and I_{CC2}.
 3. Typical values are for T_A = 25°C and nominal supply voltages.
 4. This parameter is only sampled and is not 100% tested. Output Float is defined as the point where data is no longer driven — see timing diagram on page 3.

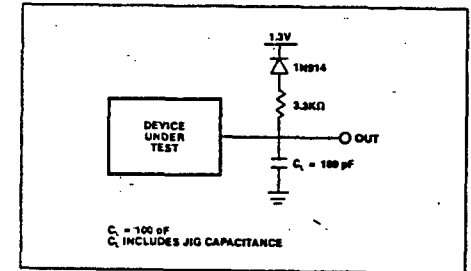
CAPACITANCE⁽²⁾ (T_A = 25°C, f = 1 MHz)

Symbol	Parameter	Typ.	Max.	Unit	Conditions
C _{IN1}	Input Capacitance Except \overline{OE}/V_{PP}	4	6	pF	V _{IN} = 0V
C _{IN2}	\overline{OE}/V_{PP} Input Capacitance		20	pF	V _{IN} = 0V
C _{OUT}	Output Capacitance	8	12	pF	V _{OUT} = 0V

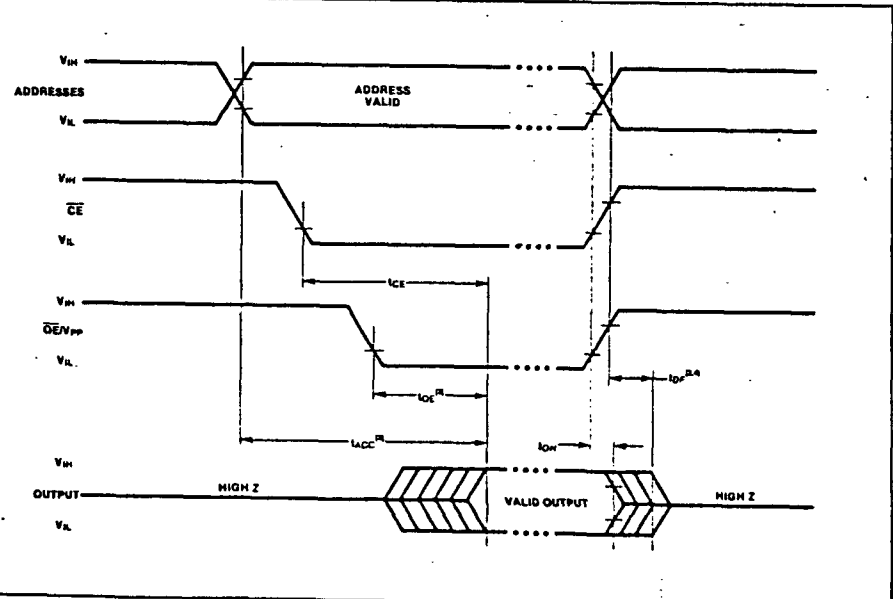
D.C. TESTING INPUT/OUTPUT WAVEFORM



A.C. TESTING LOAD CIRCUIT



A.C. WAVEFORMS



ERASURE CHARACTERISTICS

The erasure characteristics of the 2732A are such that erasure begins to occur upon exposure to light with wavelengths shorter than approximately 4000 Angstroms (Å). It should be noted that sunlight and certain types of fluorescent lamps have wavelengths in the 3000-4000 Å range. Data show that constant exposure to room level fluorescent lighting could erase the typical 2732A in approximately 3 years, while it would take approximately 1 week to cause erasure when exposed to direct sunlight. If the 2732A is to be exposed to these types of lighting conditions for extended periods of time, opaque labels should be placed over the 2732A window to prevent unintentional erasure.

The recommended erasure procedure for the 2732A is exposure to shortwave ultraviolet light which has a wavelength of 2537 Angstroms (Å). The integrated dose (i.e., UV intensity X exposure time) for erasure should be a minimum of 15 W-sec/cm². The erasure time with this dosage is approximately 15 to 20 minutes using an ultraviolet lamp with 12000µW/cm² power rating. The 2732A should be placed within 1 inch of the lamp tubes during erasure.

DEVICE OPERATION

The six modes of operation of the 2732A are listed in Table 1. A single 5V power supply is required in the read mode. All inputs are TTL levels except for \overline{OE}/V_{pp} during programming and 12V on A₉ for the intelligent Identifier™ mode. In the program mode the \overline{OE}/V_{pp} input is pulsed from a TTL level to 21V.

Table 1. Mode Selection

MODE	PINS	\overline{CE} (18)	\overline{OE}/V_{pp} (20)	A ₉ (22)	V _{CC} (24)	OUTPUTS (9-11,13-17)
Read		V _{IL}	V _{IL}	X	+5	D _{OUT}
Output Disable		V _{IL}	V _{IH}	X	+5	High Z
Standby		V _{IH}	X	X	+5	High Z
Program		V _{IL}	V _{pp}	X	+5	D _{IN}
Program Inhibit		V _{IH}	V _{pp}	X	+5	High Z
Intelligent Identifier		V _{IL}	V _{IL}	V _H	+5	Code

Notes: 1. X can be V_{IH} or V_{IL}
2. V_H = 12.0 ± 0.5V

Read Mode

The 2732A has two control functions, both of which must be logically active in order to obtain data at the outputs. Chip Enable (\overline{CE}) is the power control and should be used for device selection. Output Enable (\overline{OE}) is the output control and should be used to gate data from the output pins, independent of device selection. Assuming that addresses are stable, address access time (t_{ACC}) is equal to the delay from \overline{CE} to output (t_{CE}). Data is available at the outputs after the falling edge of \overline{OE} , assuming that \overline{CE} has been low and addresses have been stable for at least t_{ACC} - t_{OE}.

Standby Mode

The 2732A has a standby mode which reduces the maximum active current from 125 mA to 35 mA. The 2732A is placed in the standby mode by applying a TTL-high signal to the \overline{CE} input. When in standby mode, the outputs are in a high impedance state, independent of the \overline{OE} input.

Output OR-Tieing

Because EPROMs are usually used in larger memory arrays, Intel has provided a 2 line control function that accommodates this use of multiple memory connection. The two line control function allows for:

- the lowest possible memory power dissipation, and
- complete assurance that output bus contention will not occur.

To use these two control lines most efficiently, \overline{CE} (pin 18) should be decoded and used as the primary device selecting function, while \overline{OE} (pin 20) should be made a common connection to all devices in the array and connected to the READ line from the system control bus. This assures that all deselected memory devices are in their low power standby mode and that the output pins are active only when data is desired from a particular memory device.

PROGRAMMING

CAUTION: Exceeding 22V on Pin 20 (\overline{OE}/V_{pp}) will permanently damage the 2732A.

Initially, and after each erasure, all bits of the 2732A are in the "1" state. Data is introduced by selectively programming "0"s into the desired bit locations. Although only "0"s will be programmed, both "1"s and "0"s can be present in the data word. The only way to change a "0" to a "1" is by ultraviolet light erasure.

The 2732A is in the programming mode when the \overline{OE}/V_{pp} input is at 21V. It is required that a 0.1 µF capacitor be placed across \overline{OE}/V_{pp} and ground to suppress spurious voltage transients which may damage the device. The data to be programmed is applied 8 bits in parallel to the data output pins. The levels required for the address and data inputs are TTL.

When the address and data are stable, a 50 msec. active low, TTL program pulse is applied to the \overline{CE} input. A program pulse must be applied at each address location to be programmed. You can program any location at any time—either individually, sequentially, or at random. The program pulse has a maximum width of 55 msec. The 2732A must not be programmed with a DC signal applied to the \overline{CE} input.

Programming of multiple 2732As in parallel with the same data can be easily accomplished due to the simplicity of the programming requirements. Like inputs of the paralleled 2732As may be connected together when they are programmed with the same data. A low level TTL pulse applied to the \overline{CE} input programs the paralleled 2732As.

Program Inhibit

Programming of multiple 2732As in parallel with different data is also easily accomplished. Except for \overline{CE} , all like inputs (including \overline{OE}) of the parallel 2732As may be common. A TTL level program pulse applied to a 2732A's \overline{CE} input with \overline{OE}/V_{pp} at 21V will program that 2732A. A high level \overline{CE} input inhibits the other 2732As from being programmed.

Verify

A verify (Read) should be performed on the programmed bits to determine that they were correctly programmed. The verify is accomplished with \overline{OE}/V_{pp} and \overline{CE} at V_{IL}. Data should be verified t_{DP} after the falling edge of \overline{CE} .

Intelligent Identifier™ Mode

The Intelligent Identifier Mode allows the reading out of a binary code from an EPROM that will identify its manufacturer and type. This mode is intended for use by programming equipment for the purpose of automatically matching the device to be programmed with its corresponding programming algorithm. This mode is functional in the 25°C ± 5°C ambient temperature range.

To activate this mode, the programming equipment must force 11.5V to 12.5V on address line A₉ (pin 22) of the 2732A. Two identifier bytes may then be sequenced from the device outputs by toggling address line A₀ (pin 8) from V_{IL} to V_{IH}. All other address lines must be held at V_{IL} during Intelligent Identifier Mode.

Byte 0 (A₀ = V_{IL}) represents the manufacturer code and byte 1 (A₀ = V_{IH}) the device identifier code. For the Intel 2732A, these two identifier bytes are given in Table 2. All identifiers for manufacturer and device codes will possess odd parity, with the MSB (O₇) defined as the parity bit.

Table 2. 2732A Intelligent Identifier™ Bytes

Identifier	Pins	A ₀ (8)	O ₇ (17)	O ₆ (18)	O ₅ (15)	O ₄ (14)	O ₃ (13)	O ₂ (11)	O ₁ (10)	O ₀ (9)	Hex Data
Manufacturer Code		V _{IL}	1	0	0	0	1	0	0	1	89
Device Code		V _{IH}	0	0	0	0	0	0	0	1	01

Intel began manufacturing 2732As during 1982 that contained the intelligent Identifier feature. Earlier generation devices do not contain identifier information, and if erased, will respond with a "one" (V_{OH}) on each data line when operated in this mode. Programmed, preidentifier mode 2732As will respond with the current data contained in locations 0 and 1 when subjected to the intelligent Identifier operation.

System Consideration

The power switching characteristics of HMOS-E EPROMs require careful decoupling of the devices. The supply current, I_{CC}, has three segments that are of interest to the system designer—the standby current level, the active current level, and the transient current peaks that are produced by the falling and rising edges of Chip Enable. The magnitude of these transient current peaks is dependent on the output capacitive loading of the device. The associated transient voltage peaks can be suppressed by complying with Intel's Two-Line Control, as detailed in Intel's Application Note, AP-72, and by properly selected decoupling capacitors. It is recommended that a 0.1µF ceramic capacitor be used on every device between V_{CC} and GND. This should be a high frequency capacitor of low inherent inductance and should be placed as close to the device as possible. In addition, a 4.7µF bulk electrolytic capacitor should be used between V_{CC} and GND for every eight devices. The bulk capacitor should be located near where the power supply is connected to the array. The purpose of the bulk capacitor is to overcome the voltage droop caused by the inductive effects of PC board-traces.

PROGRAMMING^[4]

D.C. PROGRAMMING CHARACTERISTICS: $T_A = 25 \pm 5^\circ\text{C}$, $V_{CC} = 5\text{V} \pm 5\%$, $V_{PP} = 21\text{V} \pm 0.5\text{V}$

Symbol	Parameter	Limits			Units	Test Conditions
		Min.	Typ.	Max.		
I_{LI}	Input Current (All Inputs)			10	μA	$V_{IN} = V_{IL}$ or V_{IH}
V_{OL}	Output Low Voltage During Verify			0.45	V	$I_{OL} = 2.1\text{ mA}$
V_{OH}	Output High Voltage During Verify	2.4			V	$I_{OH} = -400\ \mu\text{A}$
I_{CC}	V_{CC} Supply Current		85	100	mA	
V_{IL}	Input Low Level (All Inputs)	-0.1		0.8	V	
V_{IH}	Input High Level (All Inputs Except $\overline{\text{OE}}V_{PP}$)	2.0		V_{CC}	V	
I_{PP}	V_{PP} Supply Current			30	mA	$\overline{\text{CE}} = V_{IL}$, $\overline{\text{OE}} = V_{PP}$
V_{ID}	Ag intelligent Identifier Voltage	11.5		12.5	V	

A.C. PROGRAMMING CHARACTERISTICS: $T_A = 25 \pm 5^\circ\text{C}$, $V_{CC} = 5\text{V} \pm 5\%$, $V_{PP} = 21\text{V} \pm 0.5\text{V}$

Symbol	Parameter	Limits			Units	Test Conditions†
		Min.	Typ.	Max.		
t_{AS}	Address Setup Time	2			μs	
t_{OES}	$\overline{\text{OE}}$ Setup Time	2			μs	
t_{DS}	Data Setup Time	2			μs	
t_{AH}	Address Hold Time	0			μs	
t_{OEH}	$\overline{\text{OE}}$ Hold Time	2			μs	
t_{DH}	Data Hold Time	2			μs	
t_{DFP}	$\overline{\text{OE}}$ High to Output Not Driven	0		130	ns	
t_{DV}	Data Valid from $\overline{\text{CE}}$			1	μs	$\overline{\text{CE}} = V_{IL}$, $\overline{\text{OE}} = V_{IL}$
t_{PW}	$\overline{\text{CE}}$ Pulse Width During Programming	20	50	55	ms	
t_{PRT}	$\overline{\text{OE}}$ Pulse Rise Time During Programming	50			ns	
t_{VR}	V_{PP} Recovery Time	2			μs	

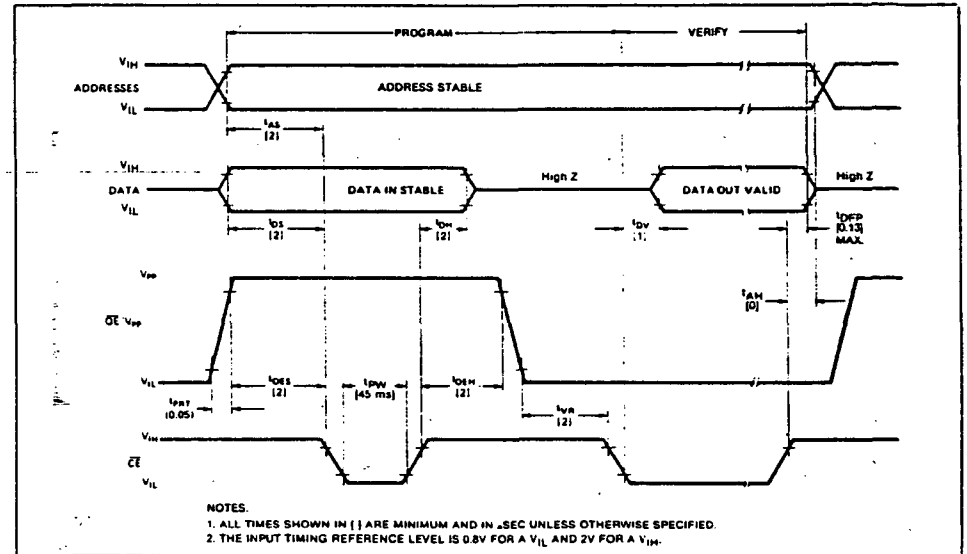
†A.C. TEST CONDITIONS

Input Rise and Fall Times (10% to 90%) $\leq 20\text{ ns}$
 Input Pulse Levels 0.45V to 2.4V
 Input Timing Reference Level 0.8V and 2.0V
 Output Timing Reference Level 0.8V and 2.0V

NOTES:

- Typical values are for $T_A = 25^\circ\text{C}$ and nominal supply voltages.
- This parameter is only sampled and is not 100% tested. Output float is defined as the point where data is no longer driven — see timing diagram.
- $\overline{\text{OE}}$ may be delayed up to $t_{ACC} - t_{OE}$ after the falling edge of $\overline{\text{CE}}$ without impacting t_{ACC} .
- When programming the 2732A, a $0.1\ \mu\text{F}$ capacitor is required across $\overline{\text{OE}}V_{PP}$ and ground to suppress spurious voltage transients which may damage the device.

PROGRAMMING WAVEFORMS



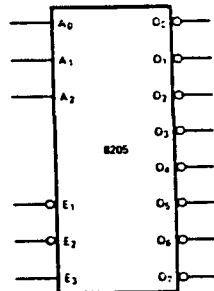
- NOTES:**
- ALL TIMES SHOWN IN [] ARE MINIMUM AND IN **MAX.** UNLESS OTHERWISE SPECIFIED.
 - THE INPUT TIMING REFERENCE LEVEL IS 0.8V FOR A V_{IL} AND 2V FOR A V_{IH} .

8205 HIGH SPEED 1 OUT OF 8 BINARY DECODER

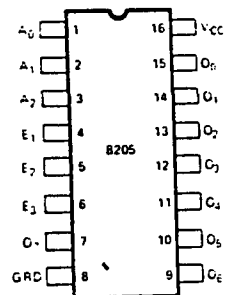
- I/O Port or Memory Selector
- Simple Expansion — Enable Inputs
- High Speed Schottky Bipolar Technology — 18ns Max. Delay
- Directly Compatible with TTL Logic Circuits
- Low Input Load Current — .25 mA max., 1/6 Standard TTL Input Load
- Minimum Line Reflection — Low Voltage Diode Input Clamp
- Outputs Sink 10 mA min.
- 16-Pin Dual-In-Line Ceramic or Plastic Package

The Intel® 8205 decoder can be used for expansion of systems which utilize input ports, output ports, and memory components with active low chip select input. When the 8205 is enabled, one of its 8 outputs goes "low," thus a single row of a memory system is selected. The 3-chip enable inputs on the 8205 allow easy system expansion. For very large systems, 8205 decoders can be cascaded such that each decoder can drive 8 other decoders for arbitrary memory expansions.

The 8205 is packaged in a standard 16-pin dual in-line package, and its performance is specified over the temperature range of 0°C to +75°C, ambient. The use of Schottky barrier diode clamped transistors to obtain fast switching speeds results in higher performance than equivalent devices made with a gold diffusion process.



ADDRESS			ENABLE			OUTPUTS							
A ₀	A ₁	A ₂	E ₁	E ₂	E ₃	0	1	2	3	4	5	6	7
L	L	L	L	L	L	L	H	H	H	H	H	H	H
L	L	L	L	L	H	L	H	H	H	H	H	H	H
L	L	L	L	H	L	L	H	H	H	H	H	H	H
L	L	L	H	L	L	L	H	H	H	H	H	H	H
L	L	H	L	L	L	L	H	H	H	H	H	H	H
L	H	L	L	L	L	L	H	H	H	H	H	H	H
L	H	L	L	L	H	L	H	H	H	H	H	H	H
L	H	L	L	H	L	L	H	H	H	H	H	H	H
L	H	L	H	L	L	L	H	H	H	H	H	H	H
L	H	L	H	L	H	L	H	H	H	H	H	H	H
L	H	L	H	H	L	L	H	H	H	H	H	H	H
L	H	L	H	H	H	L	H	H	H	H	H	H	H
L	H	H	L	L	L	L	H	H	H	H	H	H	H
L	H	H	L	L	H	L	H	H	H	H	H	H	H
L	H	H	L	H	L	L	H	H	H	H	H	H	H
L	H	H	L	H	H	L	H	H	H	H	H	H	H
L	H	H	H	L	L	L	H	H	H	H	H	H	H
L	H	H	H	L	H	L	H	H	H	H	H	H	H
L	H	H	H	H	L	L	H	H	H	H	H	H	H
L	H	H	H	H	H	L	H	H	H	H	H	H	H
H	L	L	L	L	L	L	H	H	H	H	H	H	H
H	L	L	L	L	H	L	H	H	H	H	H	H	H
H	L	L	L	H	L	L	H	H	H	H	H	H	H
H	L	L	H	L	L	L	H	H	H	H	H	H	H
H	L	L	H	L	H	L	H	H	H	H	H	H	H
H	L	L	H	H	L	L	H	H	H	H	H	H	H
H	L	L	H	H	H	L	H	H	H	H	H	H	H
H	L	H	L	L	L	L	H	H	H	H	H	H	H
H	L	H	L	L	H	L	H	H	H	H	H	H	H
H	L	H	L	H	L	L	H	H	H	H	H	H	H
H	L	H	L	H	H	L	H	H	H	H	H	H	H
H	L	H	H	L	L	L	H	H	H	H	H	H	H
H	L	H	H	L	H	L	H	H	H	H	H	H	H
H	L	H	H	H	L	L	H	H	H	H	H	H	H
H	L	H	H	H	H	L	H	H	H	H	H	H	H
H	H	L	L	L	L	L	H	H	H	H	H	H	H
H	H	L	L	L	H	L	H	H	H	H	H	H	H
H	H	L	L	H	L	L	H	H	H	H	H	H	H
H	H	L	L	H	H	L	H	H	H	H	H	H	H
H	H	L	H	L	L	L	H	H	H	H	H	H	H
H	H	L	H	L	H	L	H	H	H	H	H	H	H
H	H	L	H	H	L	L	H	H	H	H	H	H	H
H	H	L	H	H	H	L	H	H	H	H	H	H	H
H	H	H	L	L	L	L	H	H	H	H	H	H	H
H	H	H	L	L	H	L	H	H	H	H	H	H	H
H	H	H	L	H	L	L	H	H	H	H	H	H	H
H	H	H	L	H	H	L	H	H	H	H	H	H	H
H	H	H	H	L	L	L	H	H	H	H	H	H	H
H	H	H	H	L	H	L	H	H	H	H	H	H	H
H	H	H	H	H	L	L	H	H	H	H	H	H	H
H	H	H	H	H	H	L	H	H	H	H	H	H	H



A ₀ A ₁ A ₂	ADDRESS INPUTS
E ₁ E ₂ E ₃	ENABLE INPUTS
O ₀ O ₁ O ₂ O ₃ O ₄ O ₅ O ₆ O ₇	DECODED OUTPUTS

Figure 1. Logic Symbol

Figure 2. Pin Configuration

FUNCTIONAL DESCRIPTION

Decoder

The 8205 contains a one out of eight binary decoder. It accepts a three bit binary code and by gating this input, creates an exclusive output that represents the value of the input code.

For example, if a binary code of 101 was present on the A₀, A₁ and A₂ address input lines, and the device was enabled, an active low signal would appear on the O₅ output line. Note that all of the other output pins are sitting at a logic high, thus the decoded output is said to be "exclusive." The decoders outputs will follow the truth table shown below in the same manner for all other input variations.

Enable Gate

When using a decoder it is often necessary to gate the outputs with timing or enabling signals so that the exclusive output of the decoded value is synchronous with the overall system.

The 8205 has a built-in function for such gating. The three enable inputs (E₁, E₂, E₃) are ANDed together and create a single enable signal for the decoder. The combination of both active "high" and active "low" device enable inputs provides the designer with a powerfully flexible gating function to help reduce package count in his system.

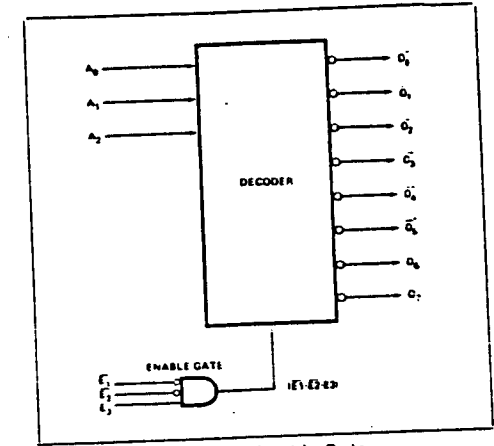


Figure 3. Enable Gate

ADDRESS			ENABLE			OUTPUTS							
A ₀	A ₁	A ₂	E ₁	E ₂	E ₃	0	1	2	3	4	5	6	7
L	L	L	L	L	L	L	H	H	H	H	H	H	H
L	L	L	L	L	H	L	H	H	H	H	H	H	H
L	L	L	L	H	L	L	H	H	H	H	H	H	H
L	L	L	H	L	L	L	H	H	H	H	H	H	H
L	L	L	H	L	H	L	H	H	H	H	H	H	H
L	L	H	L	L	L	L	H	H	H	H	H	H	H
L	L	H	L	L	H	L	H	H	H	H	H	H	H
L	L	H	L	H	L	L	H	H	H	H	H	H	H
L	L	H	L	H	H	L	H	H	H	H	H	H	H
L	L	H	H	L	L	L	H	H	H	H	H	H	H
L	L	H	H	L	H	L	H	H	H	H	H	H	H
L	L	H	H	H	L	L	H	H	H	H	H	H	H
L	L	H	H	H	H	L	H	H	H	H	H	H	H
L	H	L	L	L	L	L	H	H	H	H	H	H	H
L	H	L	L	L	H	L	H	H	H	H	H	H	H
L	H	L	L	H	L	L	H	H	H	H	H	H	H
L	H	L	L	H	H	L	H	H	H	H	H	H	H
L	H	L	H	L	L	L	H	H	H	H	H	H	H
L	H	L	H	L	H	L	H	H	H	H	H	H	H
L	H	L	H	H	L	L	H	H	H	H	H	H	H
L	H	L	H	H	H	L	H	H	H	H	H	H	H
L	H	H	L	L	L	L	H	H	H	H	H	H	H
L	H	H	L	L	H	L	H	H	H	H	H	H	H
L	H	H	L	H	L	L	H	H	H	H	H	H	H
L	H	H	L	H	H	L	H	H	H	H	H	H	H
L	H	H	H	L	L	L	H	H	H	H	H	H	H
L	H	H	H	L	H	L	H	H	H	H	H	H	H
L	H	H	H	H	L	L	H	H	H	H	H	H	H
L	H	H	H	H	H	L	H	H	H	H	H	H	H
H	L	L	L	L	L	L	H	H	H	H	H	H	H
H	L	L	L	L	H	L	H	H	H	H	H	H	H
H	L	L	L	H	L	L	H	H	H	H	H	H	H
H	L	L	L	H	H	L	H	H	H	H	H	H	H
H	L	L	H	L	L	L	H	H	H	H	H	H	H
H	L	L	H	L	H	L	H	H	H	H	H	H	H
H	L	L	H	H	L	L	H	H	H	H	H	H	H
H	L	L	H	H	H	L	H	H	H	H	H	H	H
H	L	H	L	L	L	L	H	H	H	H	H	H	H
H	L	H	L	L	H	L	H	H	H	H	H	H	H
H	L	H	L	H	L	L	H	H	H	H	H	H	H
H	L	H	L	H	H	L	H	H	H	H	H	H	H
H	L	H	H	L	L	L	H	H	H	H	H	H	H
H	L	H	H	L	H	L	H	H	H	H	H	H	H
H	L	H	H	H	L	L	H	H	H	H	H	H	H
H	L	H	H	H	H	L	H	H	H	H	H	H	H
H	H	L	L	L	L	L	H	H	H	H	H	H	H
H	H	L	L	L	H	L	H	H	H	H	H	H	H
H	H	L	L	H	L	L	H	H	H	H	H	H	H
H	H	L	L	H	H	L	H	H	H	H	H	H	H
H	H	L	H	L	L	L	H	H	H	H	H	H	H
H	H	L	H	L	H	L	H	H	H	H	H	H	H
H	H	L	H	H	L	L	H	H	H	H	H	H	H
H	H	L	H	H	H	L	H	H	H	H	H	H	H
H	H	H	L	L	L	L	H	H	H	H	H	H	H
H	H	H	L	L	H	L	H	H	H	H	H	H	H
H	H	H	L	H	L	L	H	H	H	H	H	H	H
H	H	H	L	H	H	L	H	H	H	H	H	H	H
H	H	H	H	L	L	L	H	H	H	H	H	H	H
H	H	H	H	L	H	L	H	H	H	H	H	H	H
H	H	H	H	H	L	L	H	H	H	H	H	H	H
H	H	H	H	H	H	L	H	H	H	H	H	H	H

Applications of the 8205

The 8205 can be used in a wide variety of applications in microcomputer systems. I/O ports can be decoded from the address bus and select signals can be generated to select memory devices and the type of machine state such as in 8008 systems can be derived from a simple decoding of the state lines (S1, S2) of the 8008 CPU.

I/O PORT DECODER

Shown in the figure below is a typical application of the 8205. Address rout lines are decoded by a group of 8205s (3). Each rout has a binary weight. For example, A0 is assigned a value of 1 and is the LSB; A4 is assigned a value of 16 and is the MSB. By connecting them to the decoders as shown, an active low signal that is exclusive in nature and represents the value of the input address lines, is available at the outputs of the 8205s.

This circuit can be used to generate enable signals for I/O ports or any other decoder related application.

Note that no external gating is required to decode up to 24 exclusive devices and that a simple addition of an inverter or two will allow expansion to even larger decoder networks.

CHIP SELECT DECODER

Using a very similar circuit to the I/O port decoder, an ar-

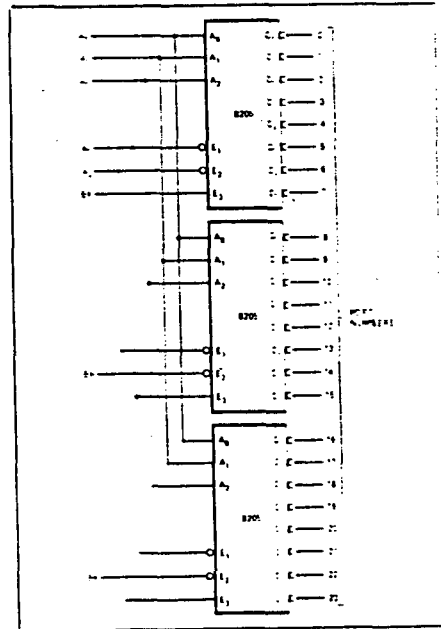


Figure 4. I/O Port Decoder

ray of 8205s can be used to stress a single channel of a 24K memory system.

The memory devices used in this application are 1K in storage capacity. The 8205s are typically used to address the memory elements and the chip select (CS) of the memory devices which come from the memory elements and the address lines of the device or group of devices connected to the output of the 8205 is active low and compatible with the memory components.

Basic operation is that the CPU places an address on a specific memory location in which it wants to "read" data. The most significant address bits are decoded by the group of 8205s and an active low chip select is generated for each memory device. The least significant address bits are used to specify location within the memory device. The address throughout the entire memory system is active low in nature and is non-redundant.

This technique can be expanded to support even larger systems with the addition of a few more 8205s and an extra decoder (8205).

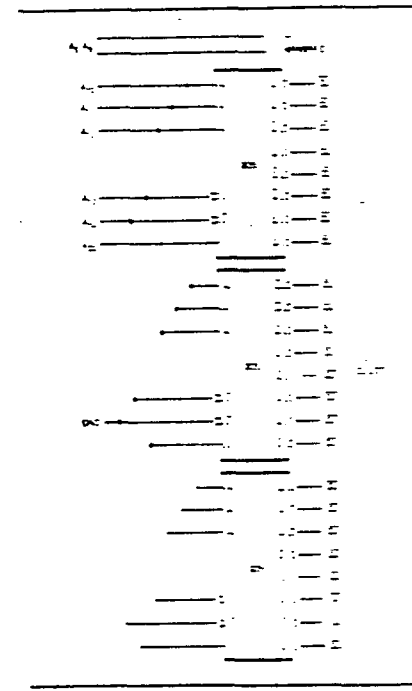


Figure 5. 24K Memory Interface

ABSOLUTE MAXIMUM RATINGS*

Temperature Under Bias:	-65°C to +125°C
Ceramic	-65°C to +75°C
Plastic	-65°C to +160°C
Storage Temperature	-0.5 to +7 Volts
Output Supply Voltages	-1.0 to +5.5 Volts
Output Currents	125 mA

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

CHARACTERISTICS (TA = 0°C to +75°C, VCC = 5V ±5%)

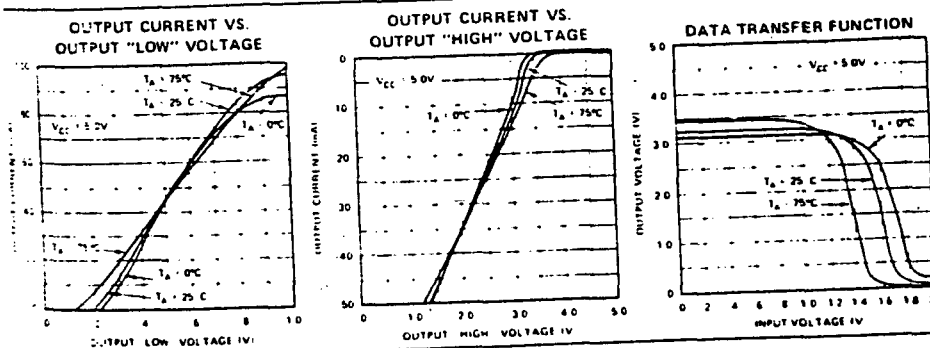
Symbol	Parameter	Limit		Unit	Test Conditions
		Min.	Max.		
II	INPUT LOAD CURRENT		-0.25	mA	VCC = 5.25V, VF = 0.45V
IL	INPUT LEAKAGE CURRENT		10	µA	VCC = 5.25V, VR = 5.25V
VIF	INPUT FORWARD CLAMP VOLTAGE		-1.0	V	VCC = 4.75V, IC = -5.0 mA
VOL	OUTPUT "LOW" VOLTAGE		0.45	V	VCC = 4.75V, IOL = 10.0 mA
VOH	OUTPUT HIGH VOLTAGE	2.4		V	VCC = 4.75V, IOH = -1.5 mA
VIL	INPUT "LOW" VOLTAGE		0.85	V	VCC = 5.0V
VIH	INPUT "HIGH" VOLTAGE	2.0		V	VCC = 5.0V
Isc	OUTPUT HIGH SHORT CIRCUIT CURRENT	-40	-120	mA	VCC = 5.0V, VOUT = 0V
VOX	OUTPUT "LOW" VOLTAGE @ HIGH CURRENT		0.8	V	VCC = 5.0V, IOX = 40 mA
ICC	POWER SUPPLY CURRENT		70	mA	VCC = 5.25V

A.C. CHARACTERISTICS (TA = 0°C to +75°C, VCC = 5V ±5%; unless otherwise specified)

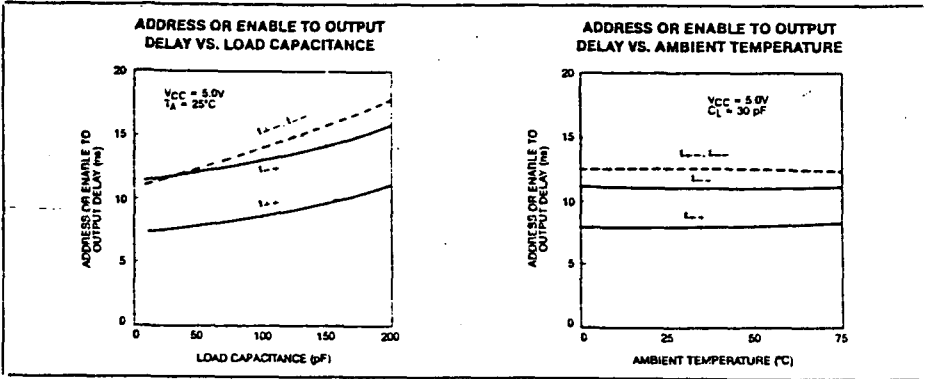
Symbol	Parameter	Max. Limit	Unit	Test Conditions
tPL	ADDRESS OR ENABLE TO OUTPUT DELAY	18	ns	f = 1 MHz, VCC = 0V, VBIAS = 2.0V, TA = 25°C
tPH		18	ns	
tOL		18	ns	
tOH		18	ns	
CIN	INPUT CAPACITANCE	4 (typ.)	pF	f = 1 MHz, VCC = 0V, VBIAS = 2.0V, TA = 25°C
		5 (typ.)	pF	

This parameter is periodically sampled and is not 100% tested.

TYPICAL CHARACTERISTICS



TYPICAL CHARACTERISTICS (Continued)

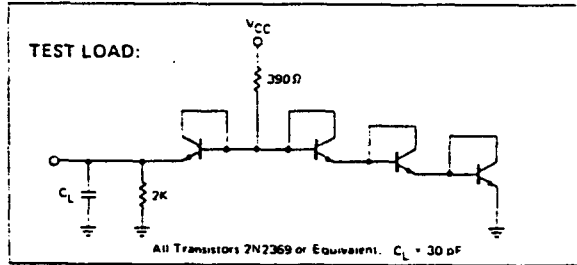


SWITCHING CHARACTERISTICS

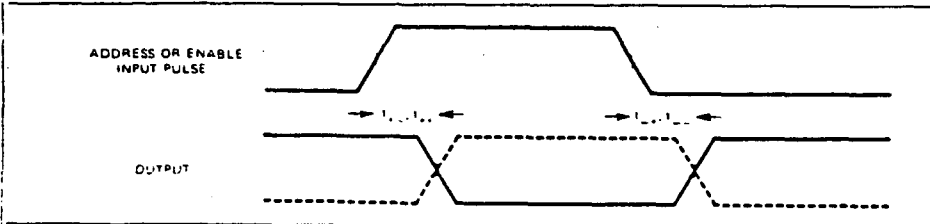
CONDITIONS OF TEST:

- Input pulse amplitudes: 2.5V
- Input rise and fall times: 5 nsec between 1V and 2V
- Measurements are made at 1.5V

TEST LOAD



WAVEFORMS



8212
8-BIT INPUT/OUTPUT PORT

- Fully Parallel 8-Bit Data Register and Buffer
- Service Request Flip-Flop for Interrupt Generation
- Low Input Load Current — .25mA Max.
- Three State Outputs
- Outputs Sink 15mA
- 3.65V Output High Voltage for Direct Interface to 8008, 8080A, or 8085A CPU
- Asynchronous Register Clear
- Replaces Buffers, Latches and Multiplexers in Microcomputer Systems
- Reduces System Package Count
- Available in EXPRESS
 - Standard Temperature Range
 - Extended Temperature Range

The 8212 input/output port consists of an 8-bit latch with 3-state output buffers along with control and device selection logic. Also included is a service request flip-flop for the generation and control of interrupts to the microprocessor.

The device is multimode in nature. It can be used to implement latches, gated buffers or multiplexers. Thus, all of the principal peripheral and input/output functions of a microcomputer system can be implemented with this device.

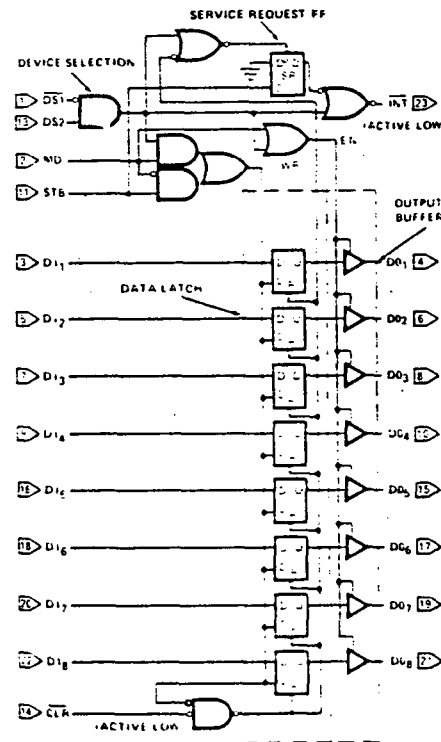


Figure 1. Logic Diagram

D ₇ , D ₆	DATA IN
D ₀ , D ₁	DATA OUT
DS ₁ , DS ₂	DEVICE SELECT
MD	MODE
STB	STROBE
INT	INTERRUPT (ACTIVE LOW)
CLR	CLEAR (ACTIVE LOW)

Figure 2. Pin Configuration

FUNCTIONAL DESCRIPTION

Data Latch

The 8 flip-flops that make up the data latch are of a "D" type design. The output (Q) of the flip-flop will follow the data input D while the clock input (C) is high. Latching will occur when the clock (C) returns low.

The latched data is cleared by an asynchronous reset input (CLR). Note: Clock (C) Overrides Reset (CLR).

Output Buffer

The outputs of the data latch (Q) are connected to 3-state, non-inverting output buffers. These buffers have a common control line EN; this control line either enables the buffer to transmit the data from the outputs of the data latch (Q) or disables the buffer, forcing the output into a high impedance state (3-state).

The high-impedance state allows the designer to connect the 8212 directly onto the microprocessor bi-directional data bus.

Control Logic

The 8212 has control inputs DS1, DS2, MD and STB. These inputs are used to control device selection, data latching, output buffer state and service request flip-flop.

DS1, DS2 (Device Select)

These 2 inputs are used for device selection. When DS1 is low and DS2 is high, DS1-DS2 the device is selected. In the selected state the output buffer is enabled and the service request flip-flop (SR) is asynchronously set.

MD (Mode)

This input is used to control the state of the output buffer and to determine the source of the clock input (C) to the data latch.

When MD is high (output mode) the output buffers are enabled and the source of clock (C) to the data latch is from the device selection logic (DS1-DS2).

When MD is low (input mode) the output buffer state is determined by the device selection logic (DS1-DS2) and the source of clock (C) to the data latch is the STB (Strobe) input.

STB (Strobe)

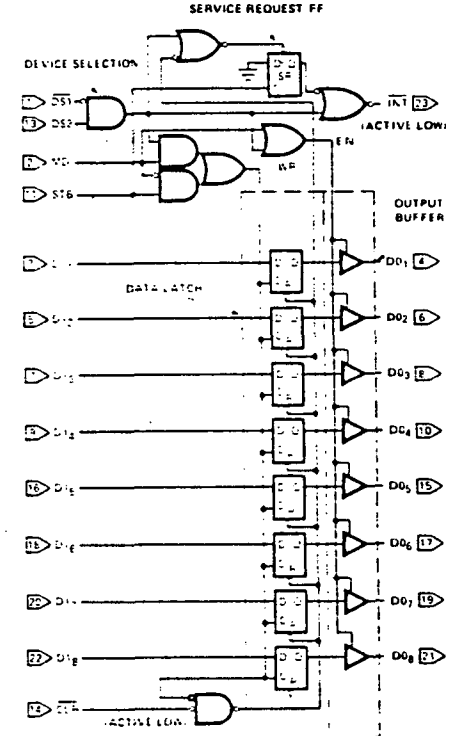
This input is used as the clock (C) to the data latch for the input mode (MD = 0) and to synchronously reset the service request flip-flop (SR).

Note that the SR flip-flop is negative edge triggered.

Service Request Flip-Flop

The (SR) flip-flop is used to generate and control interrupts in microcomputer systems. It is asynchronously set by the CLR input (active low). When the (SR) flip-flop is set it is in the non-interrupting state.

The output of the (SR) flip-flop (Q) is connected to an inverting input of a "NOR" gate. The other input to the "NOR" gate is non-inverting and is connected to the device selection logic (DS1-DS2). The output of the "NOR" gate (INT) is active low (interrupting state) for connection to active low input priority generating circuits.



STB	MD	(DS1, DS2)	DATA OUT EQUALS	CLR (DS1, DS2)	STB	"SR" INT
0	0	0	3STATE	0	0	1
1	0	0	3STATE	0	1	0
0	1	0	DATA LATCH	1	0	0
1	1	0	DATA LATCH	1	1	0
0	0	1	DATA LATCH	0	0	0
1	0	1	DATA LATCH	0	1	0
0	1	1	DATA IN	1	0	1
1	1	1	DATA IN	1	1	1

CLR - RESETS DATA LATCH
SETS SR FLIP FLOP
NO EFFECT ON OUTPUT BUFFER

INTERNAL SR FLIP FLOP

ABSOLUTE MAXIMUM RATINGS*

Temperature Under Bias Plastic	0°C to +70°C
Storage Temperature	-65°C to +160°C
All Output or Supply Voltages	-0.5 to +7 Volts
All Input Voltages	-1.0 to 5.5 Volts
Output Currents	100mA

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

D.C. CHARACTERISTICS (TA=0°C to +75°C, VCC= +5V ± 5%)

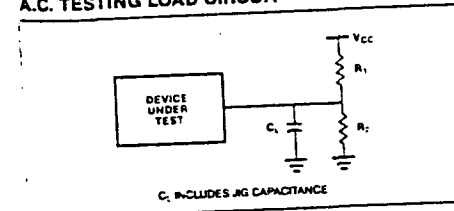
Symbol	Parameter	Limits		Unit	Test Conditions
		Min.	Typ.		
IF	Input Load Current, ACK, DS2, CR, DI1-DI8 Inputs			-25	mA V _F = .45V
IF	Input Load Current MD Input			-75	mA V _F = .45V
IF	Input Load Current DS1 Input			-1.0	mA V _F = .45V
IR	Input Leakage Current, ACK, DS, CR, DI1-DI8 Inputs			10	µA V _R ≤ V _{CC}
IR	Input Leakage Current MO Input			30	µA V _R ≤ V _{CC}
IR	Input Leakage Current DS1 Input			40	µA V _R ≤ V _{CC}
V _C	Input Forward Voltage Clamp			-1	V I _C = -5mA
V _{IL}	Input "Low" Voltage			.85	V
V _{IH}	Input "High" Voltage	2.0			V
V _{OL}	Output "Low" Voltage			.45	V I _{OL} = 15mA
V _{OH}	Output "High" Voltage	3.65	4.0		V I _{OH} = -1mA
I _{SC}	Short Circuit Output Current	-15		-75	mA V _O = 0V, V _{CC} = 5V
I _{OI}	Output Leakage Current High Impedance State			20	µA V _O = .45V/5.25V _{CC}
I _{CC}	Power Supply Current		90	130	mA

CAPACITANCE* (F = 1MHz, V_{BIAS} = 2.5V, V_{CC} = +5V, T_A = 25°C)

Symbol	Test	Limits	
		Typ.	Max.
C _{IN}	DS1 MD Input Capacitance	9pF	12pF
C _{IN}	DS2, CLR, STB, DI1-DI8 Input Capacitance	5pF	9pF
C _{OUT}	DO1-DO8 Output Capacitance	8pF	12pF

*This parameter is sampled and not 100% tested.

A.C. TESTING LOAD CIRCUIT



SWITCHING CHARACTERISTICS

Conditions of Test

Input Pulse Amplitude = 2.5V
Input Rise and Fall Times 5ns
Between 1V and 2V Measurements made at 1.5V with 15mA and 30pF Test Load

NOTE:

Test	C _L *	R ₁	R ₂
1E. ENABLE!	30pF	300Ω	600Ω
1E. DISABLE!	30pF	10KΩ	1KΩ
1E. ENABLE!	30pF	300Ω	600Ω
1E. DISABLE!	5pF	300Ω	600Ω
1E. DISABLE!	5pF	10KΩ	1KΩ

*Includes probe and µg capacitance.

A.C. CHARACTERISTICS ($T_A = 0^\circ\text{C}$ to $+70^\circ\text{C}$, $V_{CC} = +5V \pm 5\%$)

Symbol	Parameter	Limits			Unit	Test Conditions
		Min.	Typ.	Max.		
IPW	Pulse Width	30			ns	
IPD	Data to Output Delay			30	ns	Note 1
IWE	Write Enable to Output Delay			40	ns	Note 1
ISET	Data Set Up Time	15			ns	
tH	Data Hold Time	20			ns	
tR	Reset to Output Delay			40	ns	Note 1
tS	Set to Output Delay			30	ns	Note 1
tE	Output Enable/Disable Time			45	ns	Note 1
tC	Clear to Output Delay			55	ns	Note 1

*Note: For extended Temperature EXPRESS use MB212 AC Electricals Parameters.

APPLICATIONS

Basic Schematic Symbols

Two examples of ways to draw the 8212 on system schematics—(1) the top being the detailed view showing pin numbers, and (2) the bottom being the symbolic view showing the system input or output as a system bus (bus containing 8 parallel lines). The output to the data bus is symbolic in referencing 8 parallel lines.

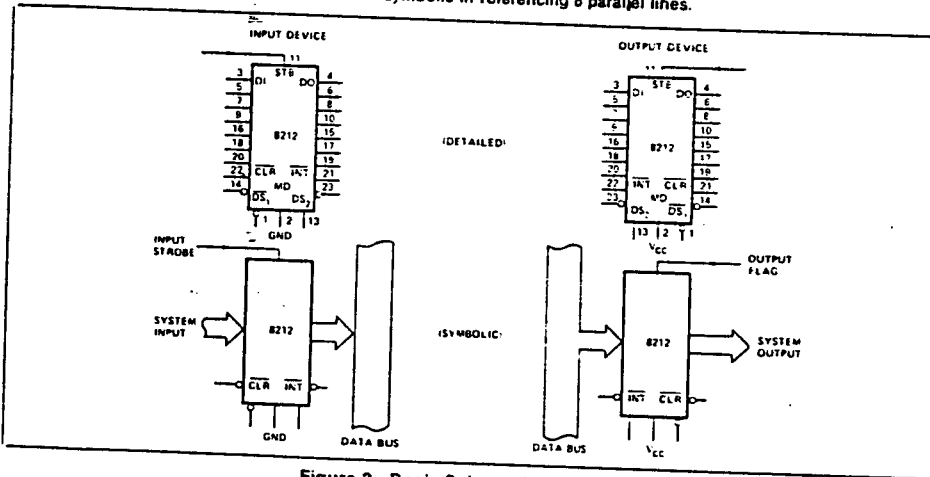


Figure 3. Basic Schematic Symbols

Gated Buffer (3-State)

The simplest use of the 8212 is that of a gated buffer. By tying the mode signal low and the strobe input high, the data latch is acting as a straight through gate. The output buffers are then enabled from the device selection logic DS1 and DS2.

When the device selection logic is false, the outputs are 3-state.

When the device selection logic is true, the input data from the system is directly transferred to the output. The input data load is 250 micro amps. The output data can sink 15 milli amps. The minimum high output is 3.65 volts.

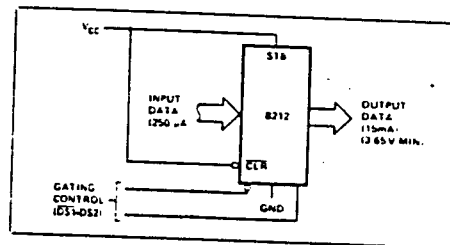


Figure 4. Gated Buffer

Bi-Directional Bus Driver

A pair of 8212's wired (back-to-back) can be used as a symmetrical drive, bi-directional bus driver. The devices are controlled by the data bus input control which is connected to DS1 on the first 8212 and to DS2 on the second. One device is active, and acting as a straight through buffer the other is in 3-state mode. This is a very useful circuit in small system design.

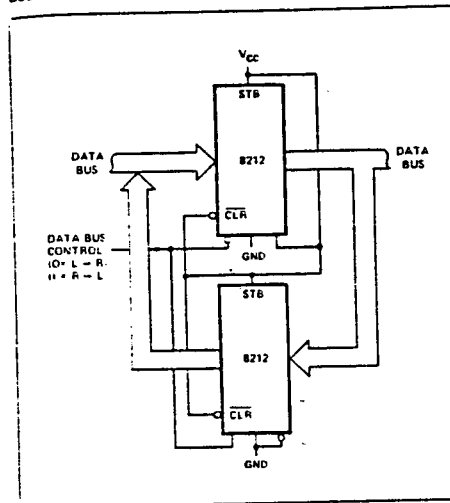


Figure 5. Bidirectional Bus Driver

Interrupting Input Port

This use of an 8212 is that of a system input port that accepts a strobe from the system input source, which in turn clears the service request flip-flop and interrupts the processor. The processor then goes through a service routine, identifies the port, and causes the device selection logic to go true — enabling the system input data onto the data bus.

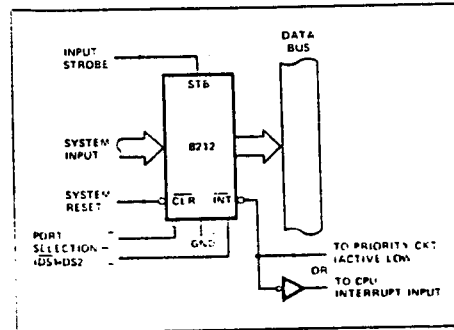


Figure 6. Interrupting Input Port

Interrupt Instruction Port

The 8212 can be used to gate the interrupt instruction, normally RESTART instructions, onto the data bus. The device is enabled from the interrupt acknowledge signal from the microprocessor and from a port selection signal. This signal is normally tied to ground. (DS1 could be used to multiplex a variety of interrupt instruction ports onto a common bus).

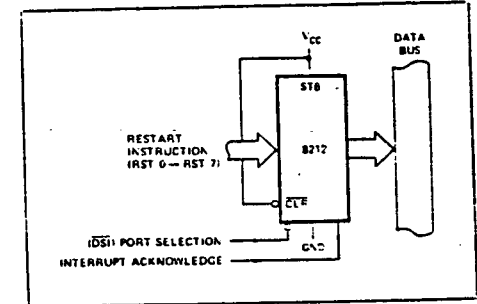


Figure 7. Interrupt Instruction Port

Output Port (With Hand-Shaking)

The 8212 can be used to transmit data from the data bus to a system output. The output strobe could be a hand-shaking signal such as "reception of data" from the device that the system is outputting to. It in turn, can interrupt the system signifying the reception of data. The selection of the port comes from the device selection logic. (DS1 · DS2)

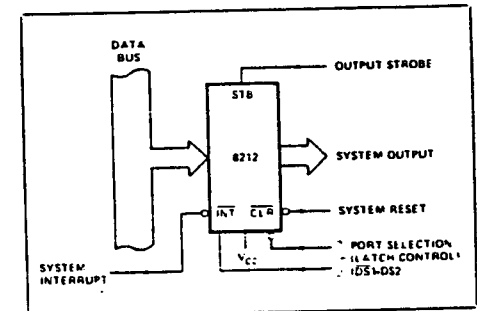
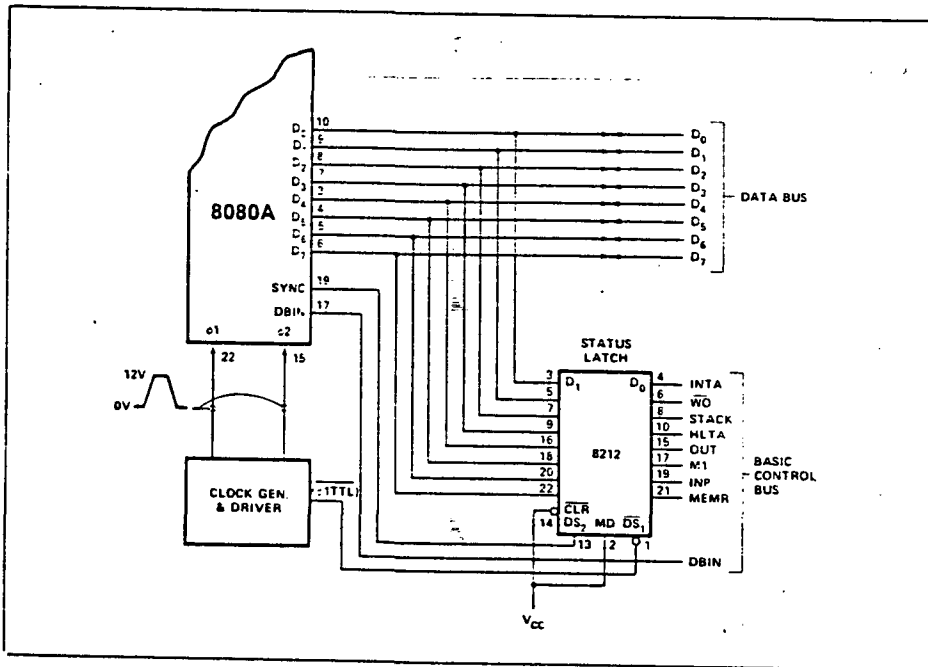
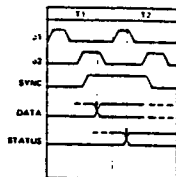


Figure 8. Output Port

808A Status Latch

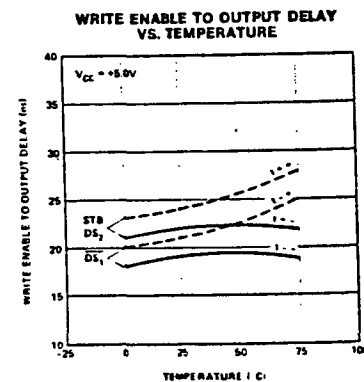
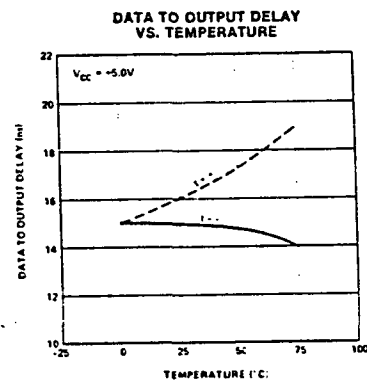
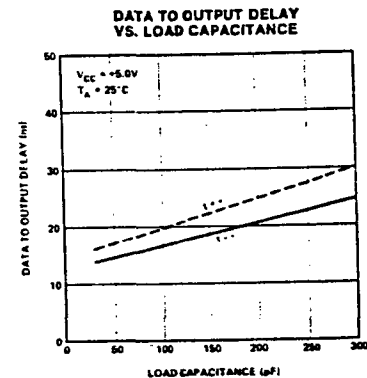
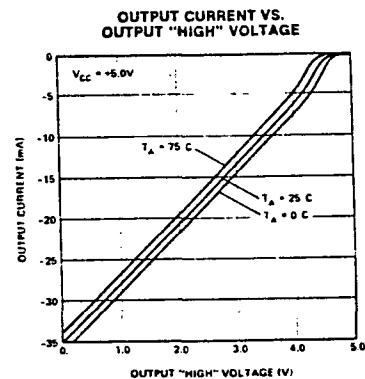
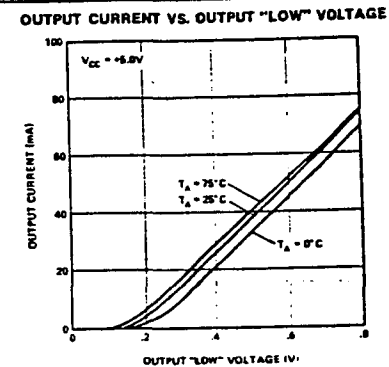
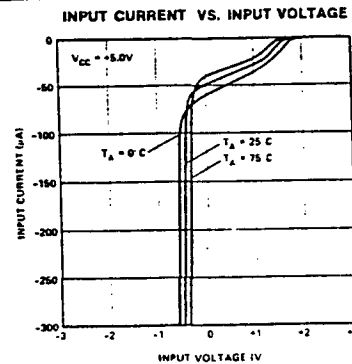
Here the 8212 is used as the status latch for an 8080A microcomputer system. The input to the 8212 latch is directly from the 8080A data bus. Timing shows that when the SYNC signal is true, which is connected to the DS2 input and the phase 1 signal is true, which is a TTL level coming from the clock generator, then, the status data will be latched into the 8212.



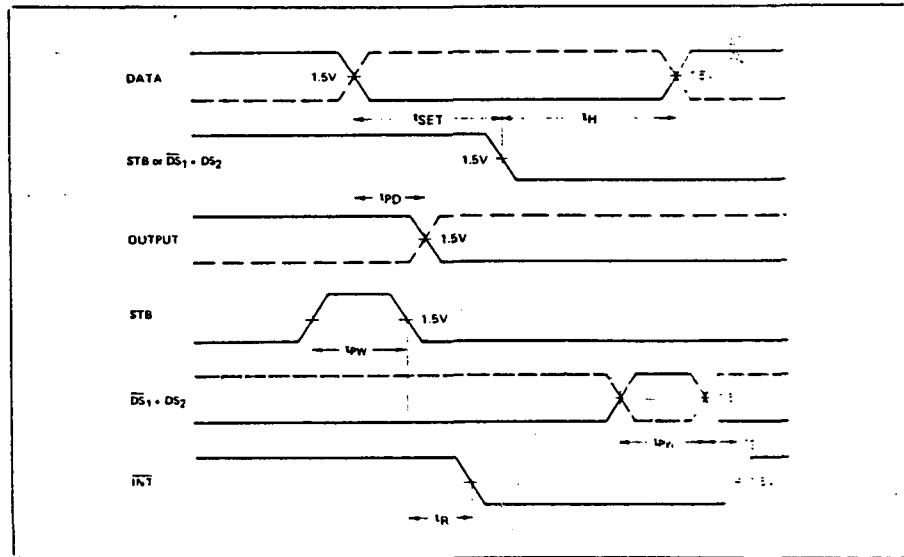
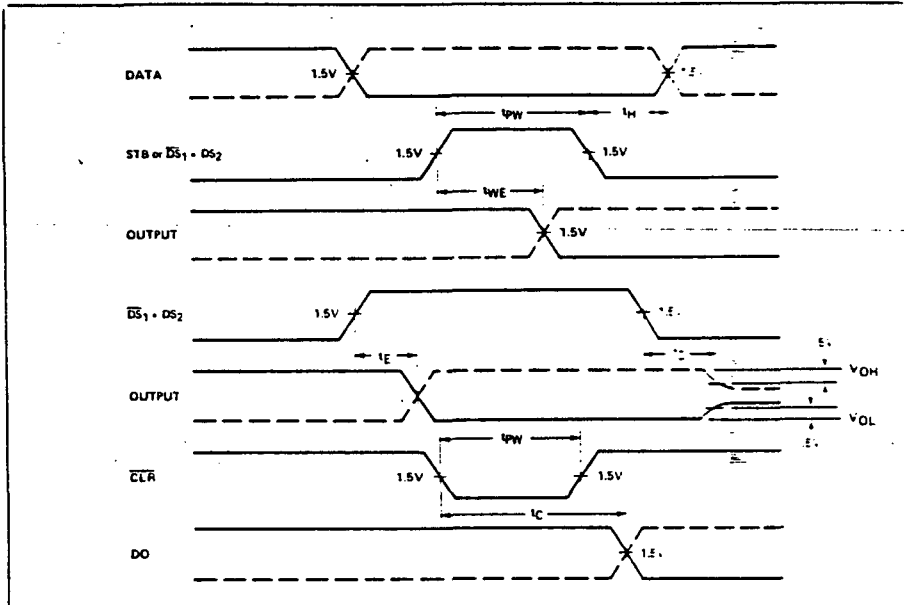
Note: The mode signal is tied high so that the output on the latch is active and enabled all the time.

It is shown that the two areas of concern are the bi-directional data bus of the microprocessor and the control bus.

TYPICAL CHARACTERISTICS



WAVEFORMS



8216/8226 4-BIT PARALLEL BIDIRECTIONAL BUS DRIVER

- Data Bus Buffer Driver for 8080 CPU
- Low Input Load Current — 0.25 mA Maximum
- High Output Drive Capability for Driving System Bus
- 3.65V Output High Voltage for Direct Interface to 8080 CPU
- 3-State Outputs
- Reduces System Package Count
- Available in EXPRESS - Standard Temperature Range

The 8216/8226 is a 4-bit bidirectional bus driver/receiver. All inputs are low power TTL compatible. For driving MOS, the DO outputs provide a high 3.65V V_{OH} , and for high capacitance terminated bus structures, the DB outputs provide a high 50 mA I_{OL} capability. A non-inverting (8216) and an inverting (8226) are available to meet a wide variety of applications for buffering in microcomputer systems.

*Note: The specifications for the 3216/3226 are identical with those for the 8216/8226.

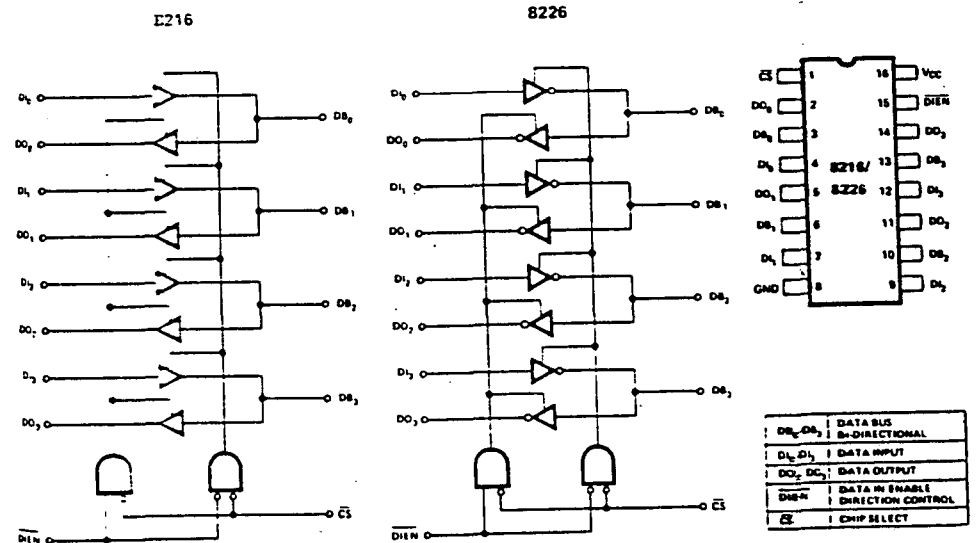
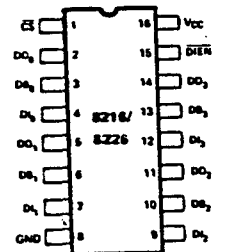


Figure 1. Block Diagrams



D ₀ , D ₁	DATA BUS BI-DIRECTIONAL
D ₂ , D ₃	DATA INPUT
DO ₀ , DO ₁	DATA OUTPUT
DIN	DATA IN ENABLE DIRECTION CONTROL
CS	CHIP SELECT

Figure 2. Pin Configuration

8085AH/8085AH-2/8085AH-1 8-BIT HMOS MICROPROCESSORS

- Single +5V Power Supply with 10% Voltage Margins
- 3 MHz, 5 MHz and 6 MHz Selections Available
- 20% Lower Power Consumption than 8085A for 3 MHz and 5 MHz
- 1.3 μ s Instruction Cycle (8085AH); 0.8 μ s (8085AH-2); 0.67 μ s (8085AH-1)
- 100% Compatible with 8085A
- 100% Software Compatible with 8080A
- On-Chip Clock Generator (with External Crystal, LC or RC Network)
- On-Chip System Controller; Advanced Cycle Status Information Available for Large System Control
- Four Vectored Interrupt Inputs (One is Non-Maskable) Plus an 8080A-Compatible Interrupt
- Serial In/Serial Out Port
- Decimal, Binary and Double Precision Arithmetic
- Direct Addressing Capability to 64K Bytes of Memory
- Available in EXPRESS
 - Standard Temperature Range
 - Extended Temperature Range

The Intel[®] 8085AH is a complete 8 bit parallel Central Processing Unit (CPU) implemented in N-channel, depletion load, silicon gate technology (HMOS). Its instruction set is 100% software compatible with the 8080A microprocessor, and it is designed to improve the present 8080A's performance by higher system speed. Its high level of system integration allows a minimum system of three IC's (8085AH (CPU), 8156H (RAM/IO) and 8355/8755A (ROM/PROM/IO)) while maintaining total system expandability. The 8085AH-2 and 8085AH-1 are faster versions of the 8085AH.

The 8085AH incorporates all of the features that the 8224 (clock generator) and 8228 (system controller) provided for the 8080A, thereby offering a high level of system integration.

The 8085AH uses a multiplexed data bus. The address is split between the 8 bit address bus and the 8 bit data bus. The on-chip address latches of 8155H/8156H/8355/8755A memory products allow a direct interface with the 8085AH.

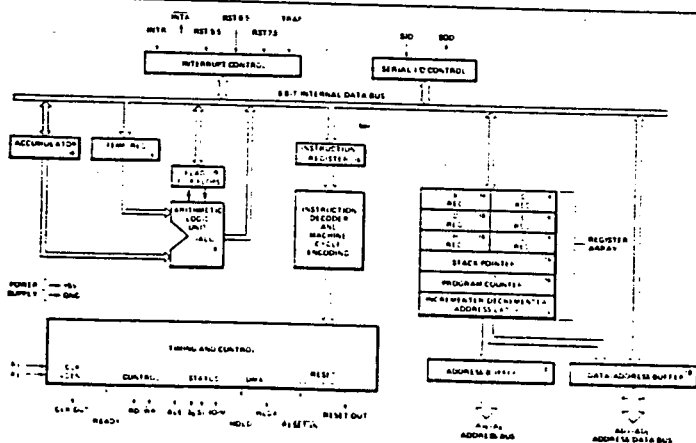


Figure 1. 8085AH CPU Functional Block Diagram

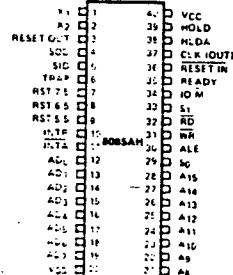


Figure 2. 8085AH Pin Configuration

Table 1. Pin Description

Symbol	Type	Name and Function	Symbol	Type	Name and Function																																												
A ₀ -A ₁₅	O	Address Bus: The most significant 8 bits of the memory address or the 8 bits of the I/O address. 3-stated during Hold and Halt modes and during RESET.	READY	I	Ready: If READY is high during a read or write cycle, it indicates that the memory or peripheral is ready to send or receive data. If READY is low, the cpu will wait an integral number of clock cycles for READY to go high before completing the read or write cycle. READY must conform to specified setup and hold times.																																												
AD ₀ -7	I/O	Multiplexed Address/Data Bus: Lower 8 bits of the memory address (or I/O address) appear on the bus during the first clock cycle (T state) of a machine cycle. It then becomes the data bus during the second and third clock cycles.	HOLD	I	Hold: Indicates that another master is requesting the use of the address and data buses. The cpu, upon receiving the hold request, will relinquish the use of the bus as soon as the completion of the current bus transfer. Internal processing can continue. The processor can regain the bus only after the HOLD is removed. When the HOLD is acknowledged, the Address, Data RD, WR, and I/O/M lines are 3-stated.																																												
ALE	O	Address Latch Enable: It occurs during the first clock state of a machine cycle and enables the address to get latched into the on-chip latch of peripherals. The falling edge of ALE is set to guarantee setup and hold times for the address information. The falling edge of ALE can also be used to strobe the status information. ALE is never 3-stated.	HLDA	O	Hold Acknowledge: Indicates that the cpu has received the HOLD request and that it will relinquish the bus in the next clock cycle. HLDA goes low after the Hold request is removed. The cpu takes the bus one half clock cycle after HLDA goes low.																																												
S ₀ , S ₁ , and IO/M	O	Machine Cycle Status: <table border="1" style="margin-left: 20px;"> <thead> <tr> <th>IO/M</th> <th>S₁</th> <th>S₀</th> <th>Status</th> </tr> </thead> <tbody> <tr><td>0</td><td>0</td><td>1</td><td>Memory write</td></tr> <tr><td>0</td><td>1</td><td>0</td><td>Memory read</td></tr> <tr><td>1</td><td>0</td><td>1</td><td>I/O write</td></tr> <tr><td>1</td><td>1</td><td>0</td><td>I/O read</td></tr> <tr><td>0</td><td>1</td><td>1</td><td>Opcode fetch</td></tr> <tr><td>1</td><td>1</td><td>1</td><td>Opcode fetch</td></tr> <tr><td>1</td><td>1</td><td>1</td><td>Interrupt Acknowledge</td></tr> <tr><td>0</td><td>0</td><td>0</td><td>Halt</td></tr> <tr><td>X</td><td>X</td><td>X</td><td>Hold</td></tr> <tr><td>X</td><td>X</td><td>X</td><td>Reset</td></tr> </tbody> </table> * = 3-state (high impedance) X = unspecified	IO/M	S ₁	S ₀	Status	0	0	1	Memory write	0	1	0	Memory read	1	0	1	I/O write	1	1	0	I/O read	0	1	1	Opcode fetch	1	1	1	Opcode fetch	1	1	1	Interrupt Acknowledge	0	0	0	Halt	X	X	X	Hold	X	X	X	Reset	INTR	I	Interrupt Request: Is used as a general purpose interrupt. It is sampled only during the next to the last clock cycle of an instruction and during Hold and Halt states. If it is active, the Program Counter (PC) will be inhibited from incrementing and an INTA will be issued. During this cycle a RESTART or CALL instruction can be inserted to jump to the interrupt service routine. The INTR is enabled and disabled by software. It is disabled by Reset and immediately after an interrupt is accepted.
IO/M	S ₁	S ₀	Status																																														
0	0	1	Memory write																																														
0	1	0	Memory read																																														
1	0	1	I/O write																																														
1	1	0	I/O read																																														
0	1	1	Opcode fetch																																														
1	1	1	Opcode fetch																																														
1	1	1	Interrupt Acknowledge																																														
0	0	0	Halt																																														
X	X	X	Hold																																														
X	X	X	Reset																																														
RD	O	Read Control: A low level on RD indicates the selected memory or I/O device is to be read and that the Data Bus is available for the data transfer. 3-stated during Hold and Halt modes and during RESET.	INTA	O	Interrupt Acknowledge: Is used instead of (and has the same timing as) RD during the instruction cycle after an INTR is accepted. It can be used to activate an 8259A interrupt chip or some other interrupt port.																																												
WR	O	Write Control: A low level on WR indicates the data on the Data Bus is to be written into the selected memory or I/O location. Data is set up at the trailing edge of WR. 3-stated during Hold and Halt modes and during RESET.	RST 5.5 RST 6.5 RST 7.5	I	Restart Interrupts: These three inputs have the same timing as INTR except they cause an internal RESTART to be automatically inserted. The priority of these interrupts is ordered as shown in Table 2. These interrupts have a higher priority than INTR. In addition, they may be individually masked out using the SIM instruction.																																												

Table 1. Pin Description (Continued)

Symbol	Type	Name and Function	Symbol	Type	Name and Function
TRAP	I	Trap: Trap interrupt is a non-maskable RESTART interrupt. It is recognized at the same time as INTR or RST 5.5-7.5. It is unaffected by any mask or interrupt Enable. It has the highest priority of any interrupt. (See Table 2.)	RESET OUT	O	ResetOut: ResetOut indicates cpu is being reset. Can be used as a system reset. The signal is synchronized to the processor clock and lasts an integral number of clock periods.
RESET IN	I	Reset In: Sets the Program Counter to zero and resets the Interrupt Enable and HLDA flip-flops. The data and address buses and the control lines are 3-stated during RESET and because of the asynchronous nature of RESET, the processor's internal registers and flags may be altered by RESET with unpredictable results. RESET IN is a Schmitt-triggered input, allowing connection to an R-C network for power-on RESET delay (see Figure 3). Upon power-up, RESET IN must remain low for at least 10 ms after minimum VCC has been reached. For proper reset operation after the power-up duration, RESET IN should be kept low a minimum of three clock periods. The CPU is held in the reset condition as long as RESET IN is applied.	X ₁ , X ₂	I	X ₁ and X ₂ : Are connected to a crystal, LC, or RC network to drive the internal clock generator. X ₁ can also be an external clock input from a logic gate. The input frequency is divided by 2 to give the processor's internal operating frequency.
			CLK	O	Clock: Clock output for use as a system clock. The period of CLK is twice the X ₁ , X ₂ input period.
			SID	I	Serial Input Data Line: The data on this line is loaded into accumulator bit 7 whenever a RIM instruction is executed.
			SOD	O	Serial Output Data Line: The output SOD is set or reset as specified by the SIM instruction.
			VCC	I	Power: +5 volt supply.
			VSS	I	Ground: Reference.

Table 2. Interrupt Priority, Restart Address, and Sensitivity

Name	Priority	Address Branched To (1) When Interrupt Occurs	Type Trigger
TRAP	1	24H	Rising edge AND high level until sampled.
RST 7.5	2	3CH	Rising edge (latched).
RST 6.5	3	34H	High level until sampled.
RST 5.5	4	2CH	High level until sampled.
INTR	5	See Note 1, 2	High level until sampled.

NOTES:

1. The processor pushes the PC on the stack before branching to the indicated address.
2. The address branched to depends on the instruction provided to the cpu when the interrupt is acknowledged.

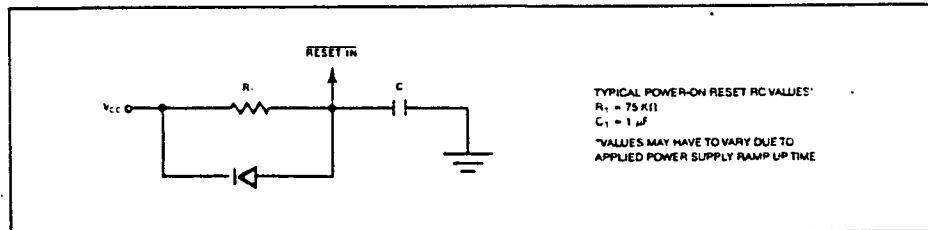


Figure 3. Power-On Reset Circuit

FUNCTIONAL DESCRIPTION

The 8085AH is a complete 8-bit parallel central processor. It is designed with N-channel, depletion load, silicon gate technology (HMOS), and requires a single +5 volt supply. Its basic clock speed is 3 MHz (8085AH), 5 MHz (8085AH-2), or 6 MHz (8085AH-1), thus improving on the present 8080A's performance with higher system speed. Also it is designed to fit into a minimum system of three IC's: The CPU (8085AH), a RAM/IO (8156H), and a ROM or EPROM/IO chip (8355 or 8755A).

The 8085AH has twelve addressable 8-bit registers. Four of them can function only as two 16-bit register pairs. Six others can be used interchangeably as 8-bit registers or as 16-bit register pairs. The 8085AH register set is as follows:

Mnemonic	Register	Contents
ACC or A	Accumulator	8 bits
PC	Program Counter	16-bit address
BC,DE,HL	General-Purpose Registers: data pointer (HL)	8 bits x 6 or 16 bits x 3
SP	Stack Pointer	16-bit address
Flags or F	Flag Register	5 flags (8-bit space)

The 8085AH uses a multiplexed Data Bus. The address is split between the higher 8-bit Address Bus and the lower 8-bit Address/Data Bus. During the first T state (clock cycle) of a machine cycle the low order address is sent out on the Address/Data bus. These lower 8 bits may be latched externally by the Address Latch Enable signal (ALE). During the rest of the machine cycle the data bus is used for memory or I/O data.

The 8085AH provides RD, WR, S₀, S₁, and IO/M signals for bus control. An Interrupt Acknowledge signal (INTA) is also provided. HOLD and all interrupts are synchronized with the processor's internal clock. The 8085AH also provides Serial Input Data (SID) and Serial Output Data (SOD) lines for simple serial interface.

In addition to these features, the 8085AH has three maskable, vector interrupt pins, one nonmaskable TRAP interrupt, and a bus vectored interrupt, INTR.

INTERRUPT AND SERIAL I/O

The 8085AH has 5 interrupt inputs: INTR, RST 5.5, RST 6.5, RST 7.5, and TRAP. INTR is identical in function to the 8080A INT. Each of the three RESTART inputs, 5.5, 6.5, and 7.5, has a programmable mask. TRAP is also a RESTART interrupt but it is nonmaskable.

The three maskable interrupts cause the internal execution of RESTART (saving the program counter in the stack and branching to the RESTART address) if the interrupts are enabled and if the interrupt mask is not set. The nonmaskable TRAP causes the internal execution of a RESTART vector independent of the state of the interrupt enable or masks. (See Table 2.)

There are two different types of inputs in the restart interrupts. RST 5.5 and RST 6.5 are high level-sensitive like INTR (and INT on the 8080) and are recognized with the same timing as INTR. RST 7.5 is rising edge-sensitive.

For RST 7.5, only a pulse is required to set an internal flip-flop which generates the internal interrupt request (a normally high level signal with a low going pulse is recommended for highest system noise immunity). The RST 7.5 request flip-flop remains set until the request is serviced. Then it is reset automatically. This flip-flop may also be reset by using the SIM instruction or by issuing a RESET IN to the 8085AH. The RST 7.5 internal flip-flop will be set by a pulse on the RST 7.5 pin even when the RST 7.5 interrupt is masked out.

The status of the three RST interrupt masks can only be affected by the SIM instruction and RESET IN. (See SIM, Chapter 5 of the MCS-80/85 User's Manual.)

The interrupts are arranged in a fixed priority that determines which interrupt is to be recognized if more than one is pending as follows: TRAP—highest priority, RST 7.5, RST 6.5, RST 5.5, INTR—lowest priority. This priority scheme does not take into account the priority of a routine that was started by a higher priority interrupt. RST 5.5 can interrupt an RST 7.5 routine if the interrupts are re-enabled before the end of the RST 7.5 routine.

The TRAP interrupt is useful for catastrophic events such as power failure or bus error. The TRAP input is recognized just as any other interrupt but has the highest priority. It is not affected by any flag or mask. The TRAP input is both edge and level sensitive. The TRAP input must go high and remain high until it is acknowledged. It will not be recognized again until it goes low, then high again. This avoids any false triggering due to noise or logic glitches. Figure 4 illustrates the TRAP interrupt request circuitry within the 8085AH. Note that the servicing of any interrupt (TRAP, RST 7.5, RST 6.5, RST 5.5, INTR) disables all future interrupts (except TRAPs) until an EI instruction is executed.

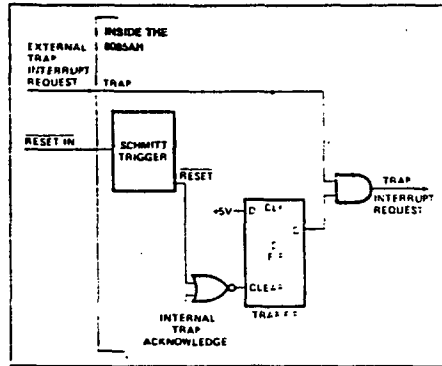


Figure 4. TRAP and RESET IN Circuit

The TRAP interrupt is special in that it disables interrupts, but preserves the previous interrupt enable status. Performing the first RIM instruction following a TRAP interrupt allows you to determine whether interrupts were enabled or disabled prior to the TRAP. All subsequent RIM instructions provide current interrupt enable status. Performing a RIM instruction following INTR, or RST 5.5-7.5 will provide current interrupt enable status, revealing that interrupts are disabled. See the description of the RIM instruction in the MCS-80/85 Family User's Manual.

The serial I/O system is also controlled by the RIM and SIM instructions. SID is read by RIM, and SIM sets the SOD data.

DRIVING THE X₁ AND X₂ INPUTS

You may drive the clock inputs of the 8085AH, 8085AH-2, or 8085AH-1 with a crystal, an LC tuned circuit, an RC network, or an external clock source. The crystal frequency must be at least 1 MHz, and must be twice the desired internal clock frequency; hence, the 8085AH is operated with a 6 MHz crystal (for 3 MHz clock), the 8085AH-2 operated with a 10 MHz crystal (for 5 MHz clock), and the 8085AH-1 can be operated with a 12 MHz crystal (for 6 MHz clock). If a crystal is used, it must have the following characteristics:

- Parallel resonance at twice the clock frequency desired
- C_L (load capacitance) ≤ 30 pF
- C_S (shunt capacitance) ≤ 7 pF
- R_S (equivalent shunt resistance) ≤ 75 Ohms
- Drive level: 10 mW
- Frequency tolerance: ±.005% (suggested)

Note the use of the 20 pF capacitor between X₂ and ground. This capacitor is required with crystal frequencies below 4 MHz to assure oscillator startup at the correct frequency. A parallel-resonant LC circuit may be used as the frequency-determining network for the 8085AH, providing that its frequency tolerance of approximately ±10% is acceptable. The components are chosen from the formula:

$$f = \frac{1}{2\pi\sqrt{L(C_{ext} + C_{int})}}$$

To minimize variations in frequency, it is recommended that you choose a value for C_{ext} that is at least twice that of C_{int}, or 30 pF. The use of an LC circuit is not recommended for frequencies higher than approximately 5 MHz.

An RC circuit may be used as the frequency-determining network for the 8085AH if maintaining a precise clock frequency is of no importance. Variations in the on-chip timing generation can cause a wide variation in frequency when using the RC mode. Its advantage is its low component cost. The driving frequency generated by the circuit shown is approximately 3 MHz. It is not recommended that frequencies greatly higher or lower than this be attempted.

Figure 5 shows the recommended clock driver circuits. Note in D and E that pullup resistors are required to assure that the high level voltage of the input is at least 4V and maximum low level voltage of 0.8V.

For driving frequencies up to and including 6 MHz you may supply the driving signal to X₁ and leave X₂ open-circuited (Figure 5D). If the driving frequency is from 6 MHz to 12 MHz, stability of the clock generator will be improved by driving both X₁ and X₂ with a push-pull source (Figure 5E). To prevent self-oscillation of the 8085AH, be sure that X₂ is not coupled back to X₁ through the driving circuit.

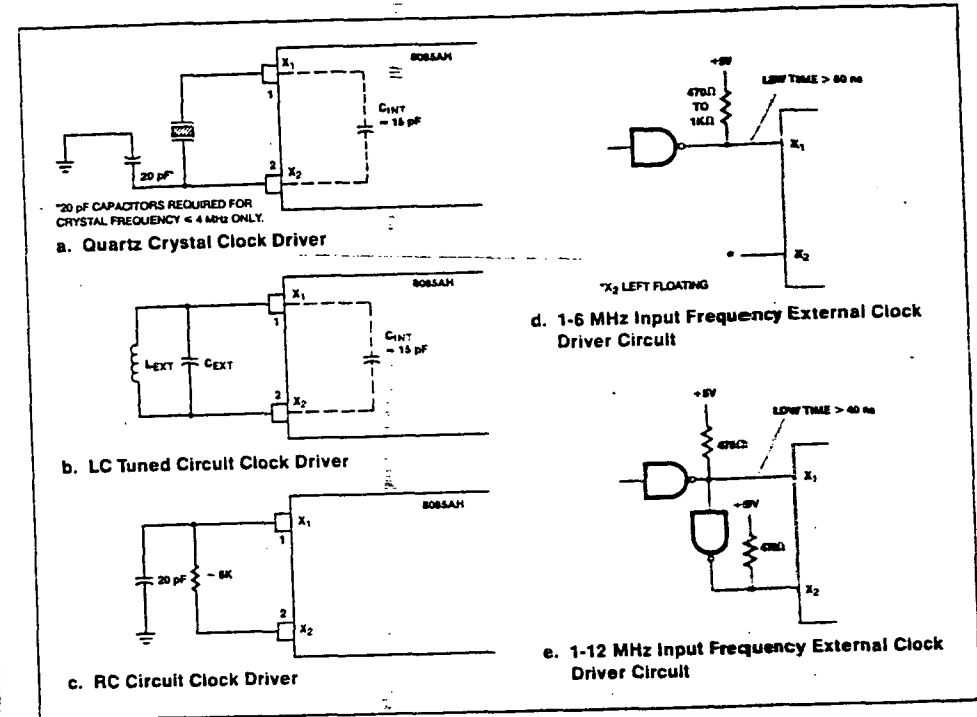


Figure 5. Clock Driver Circuits

GENERATING AN 8085AH WAIT STATE

If your system requirements are such that slow memories or peripheral devices are being used, the circuit shown in Figure 6 may be used to insert one WAIT state in each 8085AH machine cycle.

- The D flip-flops should be chosen so that
- CLK is rising edge-triggered
- CLEAR is low-level active.

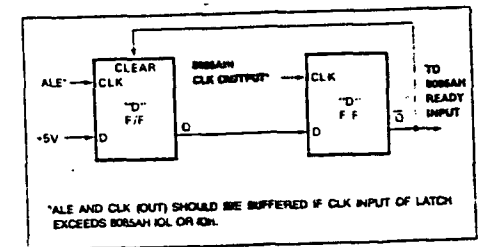


Figure 6. Generation of a Wait State for 8085AH CPU

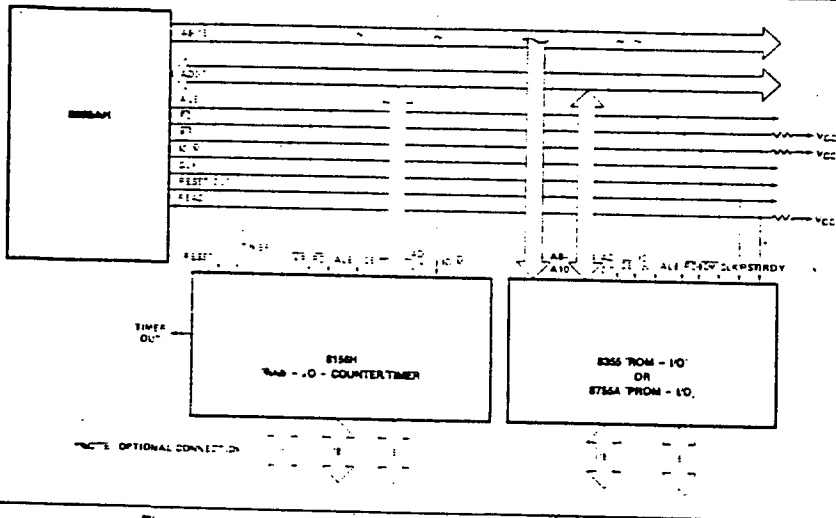


Figure 8. MCS-85 Minimum System (Memory Mapped I/O)

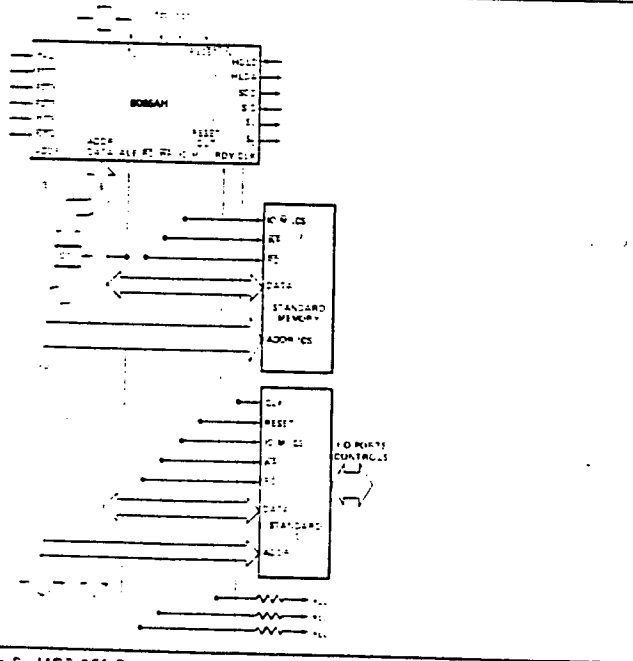


Figure 9. MCS-85 System (Using Standard Memories)

As in the 8080, the READY line is used to extend the read and write pulse lengths so that the 8085AH can be used with slow memory. HOLD causes the CPU to relinquish the bus when it is through with it by floating the Address and Data Buses.

SYSTEM INTERFACE

The 8085AH family includes memory components, which are directly compatible to the 8085AH CPU. For example, a system consisting of the three chips, 8085AH, 8156H, and 8355 will have the following features:

- 2K Bytes ROM
- 256 Bytes RAM
- 1 Timer/Counter
- 4 8-bit I/O Ports
- 1 6-bit I/O Port
- 4 Interrupt Levels
- Serial In/Serial Out Ports

This minimum system, using the standard I/O technique is as shown in Figure 7.

In addition to standard I/O, the memory mapped I/O offers an efficient I/O addressing technique. With this technique, an area of memory address space is assigned for I/O address, thereby, using the memory address for I/O manipulation. Figure 8 shows the system configuration of Memory Mapped I/O using 8085AH.

The 8085AH CPU can also interface with the standard memory that does *not* have the multiplexed address/data bus. It will require a simple 8212 (8-bit latch) as shown in Figure 9.

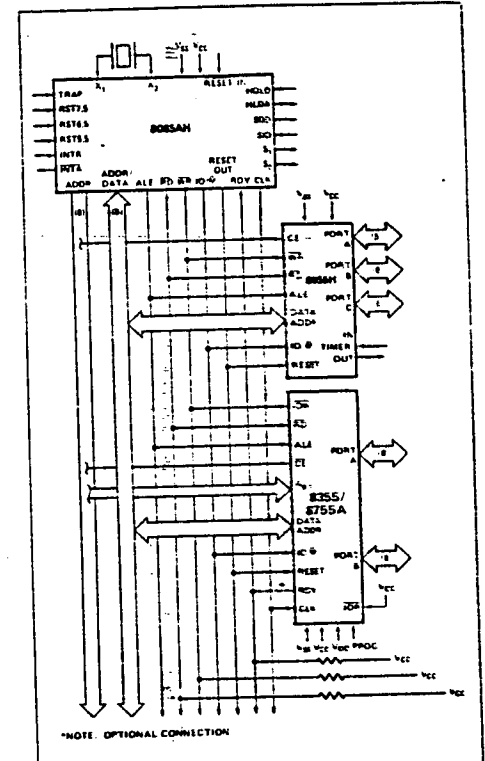


Figure 7. 8085AH Minimum System (Standard I/O Technique)

BASIC SYSTEM TIMING

The 8085AH has a multiplexed Data Bus. ALE is used as a strobe to sample the lower 8-bits of address on the Data Bus. Figure 10 shows an instruction fetch, memory read and I/O write cycle (as would occur during processing of the OUT instruction). Note that during the I/O write and read cycle that the I/O port address is copied on both the upper and lower half of the address.

There are seven possible types of machine cycles. Which of these seven takes place is defined by the status of the three status lines (IO/M, S₁, S₀) and the three control signals (RD, WR, and INTA). (See Table 3.) The status lines can be used as advanced controls (for device selection, for example), since they become active at the T₁ state, at the outset of each machine cycle. Control lines RD and WR become active later, at the time when the transfer of data is to take place, so are used as command lines.

A machine cycle normally consists of three T states, with the exception of OP CODE FETCH, which normally has either four or six T states (unless WAIT or HOLD states are forced by the receipt of READY or HOLD inputs). Any T state must be one of ten possible states, shown in Table 4.

Table 3. 8085AH Machine Cycle Chart

MACHINE CYCLE	STATUS			CONTROL		
	IO/M	S ₁	S ₀	RD	WR	INTA
OPCODE FETCH (OP)	0	1	1	0	1	1
MEMORY READ (MR)	0	1	0	0	1	1
MEMORY WRITE (MW)	0	0	1	0	1	0
I/O READ (IOR)	1	1	0	0	1	1
I/O WRITE (IOW)	1	0	1	0	1	0
ACKNOWLEDGE OF INTF	INTA	1	1	1	1	0
BUS IDLE	IBI: DAD	0	1	0	1	1
	ACK OF RST. TRAP	1	1	1	1	1
	HALT	TS	0	0	TS	TS

Table 4. 8085AH Machine State Chart

Machine State	Status & Buses					Control		
	S ₁ S ₀	IO/M	A ₈ -A ₁₅	A ₀ -A ₇	RD	WR	INTA	ALE
T ₁	X	X	X	X	1	1	1	1
T ₂	X	X	X	X	X	X	0	0
T _{WAIT}	X	X	X	X	X	X	0	0
T ₃	X	X	X	X	X	X	0	0
T ₄	1	0	X	TS	1	1	0	0
T ₅	1	0	X	TS	1	1	0	0
T ₆	1	0	Y	TS	1	1	0	0
T _{RESET}	X	TS	TS	TS	TS	1	0	0
T _{HALT}	0	TS	TS	TS	TS	1	0	0
T _{HOLD}	X	TS	TS	TS	TS	1	0	0

0 = Logic 0; 1 = Logic 1; TS = high impedance; X = unspecified.
 * ALE is generated during 2nd and 3rd machine cycles of DAD instruction.
 † IO/M = 1 during T₄-T₆ of the A machine cycle.

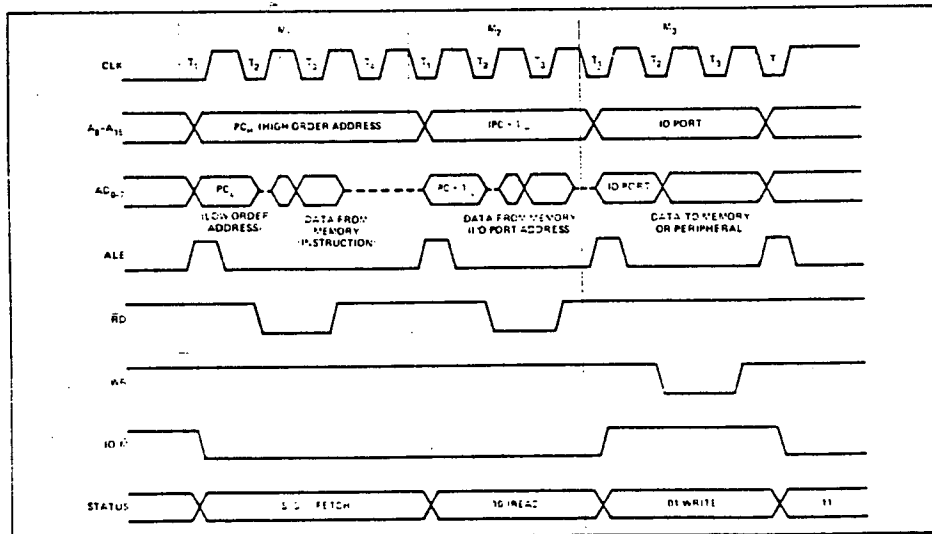


Figure 10. 8085AH Basic System Timing

ABSOLUTE MAXIMUM RATINGS*

Ambient Temperature Under Bias 0°C to 70°C
 Storage Temperature -65°C to +150°C
 Voltage on Any Pin
 With Respect to Ground -0.5V to +7V
 Power Dissipation 1.5 Watt

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

D.C. CHARACTERISTICS

8085AH, 8085AH-2: (T_A = 0°C to 70°C, V_{CC} = 5V ±10%, V_{SS} = 0V; unless otherwise specified)
 8085AH-1: (T_A = 0°C to 70°C, V_{CC} = 5V ±5%, V_{SS} = 0V; unless otherwise specified)

Symbol	Parameter	Min.	Max.	Units	Test Conditions
V _{IL}	Input Low Voltage	-0.5	-0.8	V	
V _{IH}	Input High Voltage	2.0	V _{CC} + 0.5	V	
V _{OL}	Output Low Voltage		0.45	V	I _{OL} = 2mA
V _{OH}	Output High Voltage	2.4		V	I _{OH} = -400µA
I _{CC}	Power Supply Current		135	mA	8085AH, 8085AH-2
			200	mA	8085AH-1 (Preliminary)
I _{IL}	Input Leakage		±10	µA	0 ≤ V _{IN} ≤ V _{CC}
I _{LO}	Output Leakage		±10	µA	0.45V ≤ V _{OUT} ≤ V _{CC}
V _{ILR}	Input Low Level, RESET	-0.5	+0.8	V	
V _{IHR}	Input High Level, RESET	2.4	V _{CC} + 0.5	V	
V _{HV}	Hysteresis, RESET	0.15		V	

A.C. CHARACTERISTICS

8085AH, 8085AH-2: (T_A = 0°C to 70°C, V_{CC} = 5V ±10%, V_{SS} = 0V)
 8085AH-1: (T_A = 0°C to 70°C, V_{CC} = 5V ±5%, V_{SS} = 0V)

Symbol	Parameter	8085AH ⁽²⁾ (Final)		8085AH-2 ⁽²⁾ (Final)		8085AH-1 (Preliminary)		Units
		Min.	Max.	Min.	Max.	Min.	Max.	
t _{CYC}	CLK Cycle Period	320	2000	200	2000	167	2000	ns
t ₁	CLK Low Time (Standard CLK Loading)	80		40		20		ns
t ₂	CLK High Time (Standard CLK Loading)	120		70		50		ns
t _r , t _f	CLK Rise and Fall Time		30		30		30	ns
t _{XKR}	X ₁ Rising to CLK Rising	20	120	20	100	20	100	ns
t _{XKF}	X ₁ Rising to CLK Falling	20	150	20	110	20	110	ns
t _{AC}	A ₈₋₁₅ Valid to Leading Edge of Control ⁽¹⁾	270		115		70		ns
t _{ACL}	A ₀₋₇ Valid to Leading Edge of Control	240		115		60		ns
t _{AD}	A ₀₋₁₅ Valid to Valid Data In		575		350		225	ns
t _{AFR}	Address Float After Leading Edge of READ (INTA)		0		0		0	ns
t _{AL}	A ₈₋₁₅ Valid Before Trailing Edge of ALE ⁽¹⁾	115		50		25		ns

*Note: For Extended Temperature EXPRESS use M8085AH Electricals Parameters.

A.C. CHARACTERISTICS (Continued)

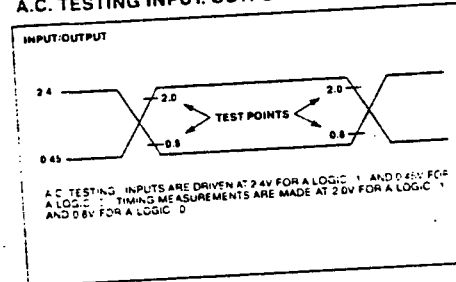
Symbol	Parameter	8085AH ^[2] (Final)		8085AH-2 ^[2] (Final)		8085AH-1 (Preliminary)		Units
		Min.	Max.	Min.	Max.	Min.	Max.	
t _{ALL}	A ₀₋₇ Valid Before Trailing Edge of ALE	90		50		25		ns
t _{ARY}	READY Valid from Address Valid		220		100		40	ns
t _{CA}	Address (A ₀₋₁₅) Valid After Control	120		60		30		ns
t _{CC}	Width of Control Low (RD, WR, INTA Edge of ALE	400		230		150		ns
t _{CL}	Trailing Edge of Control to Leading Edge of ALE	50		25		0		ns
t _{DW}	Data Valid to Trailing Edge of WRITE	420		230		140		ns
t _{HABE}	HLDA to Bus Enable		210		150		150	ns
t _{HABF}	Bus Float After HLDA		210		150		150	ns
t _{HACK}	HLDA Valid to Trailing Edge of CLK	110		40		0		ns
t _{HDH}	HOLD Hold Time	0		0		0		ns
t _{HDS}	HOLD Setup Time to Trailing Edge of CLK	170		120		120		ns
t _{INH}	INTR Hold Time	0		0		0		ns
t _{INS}	INTR, RST, and TRAP Setup Time to Falling Edge of CLK	160		150		150		ns
t _{LA}	Address Hold Time After ALE	100		50		20		ns
t _{LC}	Trailing Edge of ALE to Leading Edge of Control	130		60		25		ns
t _{LCK}	ALE Low During CLK High	100		50		15		ns
t _{LDR}	ALE to Valid Data During Read		460		270		175	ns
t _{LDW}	ALE to Valid Data During Write		200		140		110	ns
t _{LL}	ALE Width	140		80		50		ns
t _{LRV}	ALE to READY Stable		110		30		10	ns
t _{RAE}	Trailing Edge of READ to Re-Enabling of Address	150		90		50		ns
t _{RD}	READ (or INTA) to Valid Data		300		150		75	ns
t _{RV}	Control Trailing Edge to Leading Edge of Next Control	400		220		160		ns
t _{RDH}	Data Hold Time After READ INTA	0		0		0		ns
t _{RYH}	READY Hold Time	0		0		5		ns
t _{RYS}	READY Setup Time to Leading Edge of CLK	110		100		100		ns
t _{WD}	Data Valid After Trailing Edge of WRITE	100		60		30		ns
t _{WDL}	LEADING Edge of WRITE to Data valid		40		20		30	ns

NOTES:

- A₀-A₁₅ address Specs apply IO/M, S₀, and S₁ except A₀-A₁₅ are undefined during T₄-T₆ of OF cycle whereas IO/M, S₀, and S₁ are stable.
- Test Conditions: t_{CC} = 320 ns (8085AH)/200 ns (8085AH-2)/167 ns (8085AH-1); C_L = 150 pF.

- For all output timing where C_L ≠ 150 pF use the following correction factors:
 25 pF < C_L < 150 pF: -0.10 ns/pF
 150 pF < C_L < 300 pF: +0.30 ns/pF
- Output timings are measured with purely capacitive load.
- To calculate timing specifications at other values of t_{CC} use Table 5.

A.C. TESTING INPUT, OUTPUT WAVEFORM



A.C. TESTING LOAD CIRCUIT

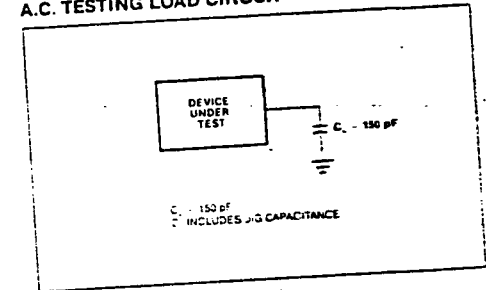


Table 5. Bus Timing Specification as a T_{CC} Dependent

Symbol	8085AH	8085AH-2	8085AH-1	
t _{AL}	(1/2) T - 45	(1/2) T - 50	(1/2) T - 58	Minimum
t _{LA}	(1/2) T - 60	(1/2) T - 50	(1/2) T - 63	Minimum
t _{LL}	(1/2) T - 20	(1/2) T - 20	(1/2) T - 33	Minimum
t _{LCK}	(1/2) T - 60	(1/2) T - 50	(1/2) T - 68	Minimum
t _{LC}	(1/2) T - 30	(1/2) T - 40	(1/2) T - 58	Minimum
t _{AD}	(5/2 + N) T - 225	(5/2 + N) T - 150	(5/2 + N) T - 192	Maximum
t _{RD}	(3/2 + N) T - 180	(3/2 + N) T - 150	(3/2 + N) T - 175	Maximum
t _{RAE}	(1/2) T - 10	(1/2) T - 10	(1/2) T - 33	Minimum
t _{CA}	(1/2) T - 40	(1/2) T - 40	(1/2) T - 53	Minimum
t _{DW}	(3/2 + N) T - 60	(3/2 - N) T - 70	(3/2 + N) T - 110	Minimum
t _{WD}	(1/2) T - 60	(1/2) T - 40	(1/2) T - 53	Minimum
t _{CC}	(3/2 + N) T - 80	(3/2 - N) T - 70	(3/2 + N) T - 100	Minimum
t _{CL}	(1/2) T - 110	(1/2) T - 75	(1/2) T - 83	Minimum
t _{ARY}	(3/2) T - 260	(3/2) T - 200	(3/2) T - 210	Maximum
t _{HACK}	(1/2) T - 50	(1/2) T - 60	(1/2) T - 83	Minimum
t _{HABF}	(1/2) T + 50	(1/2) T - 50	(1/2) T - 67	Maximum
t _{HABE}	(1/2) T + 50	(1/2) T - 50	(1/2) T - 67	Maximum
t _{HBE}	(2/2) T - 50	(2/2) T - 65	(2/2) T - 97	Maximum
t ₁	(1/2) T - 80	(1/2) T - 60	(1/2) T - 63	Minimum
t ₂	(1/2) T - 40	(1/2) T - 30	(1/2) T - 33	Minimum
t _{RV}	(3/2) T - 80	(3/2) T - 80	(3/2) T - 90	Minimum
t _{LDR}	(4/2) T - 180	(4/2) T - 130	(4/2) T - 159	Maximum

NOTE: N is equal to the total WAIT states. T = t_{CC}.

WAVEFORMS (Continued)

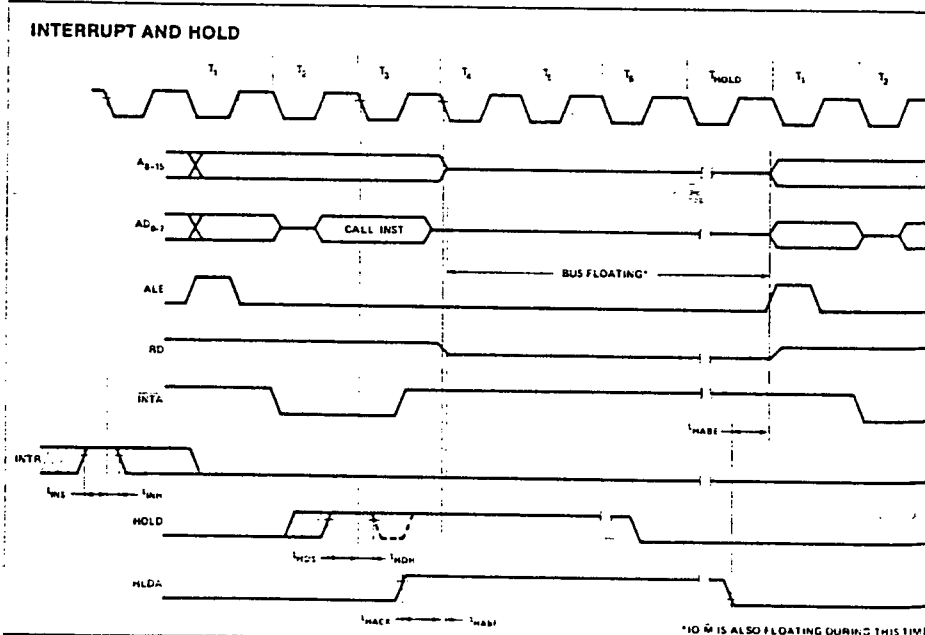
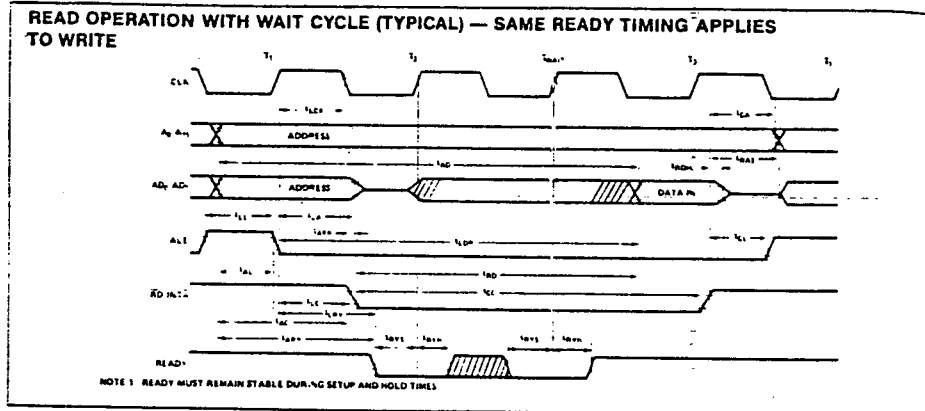


Table 6. Instruction Set Summary

Mnemonic	Instruction Code								Operations Description
	D ₇	D ₆	D ₅	D ₄	D ₃	D ₂	D ₁	D ₀	
MOVE, LOAD, AND STORE									
MOV R1 R2	0	1	D	D	D	S	S	S	Move register to register
MOV M1	0	1	1	0	S	S	S	S	Move register to memory
MOV R1 M	0	1	D	D	D	1	1	0	Move memory to register
MVI R1	0	0	D	D	D	1	1	0	Move immediate register
MVI M	0	0	D	D	D	1	1	0	Move immediate memory
LDI B	0	0	0	0	0	0	0	1	Load immediate register
LXI D	0	0	0	1	0	0	0	1	Load immediate register Pair B & C
LXI H	0	0	1	0	0	0	0	1	Load immediate register Pair D & E
STAX B	0	0	0	0	0	0	1	0	Store A indirect
STAX D	0	0	0	1	0	0	1	0	Store A indirect
LDAX B	0	0	0	0	1	0	1	0	Load A indirect
LDAX D	0	0	0	1	1	0	1	0	Load A indirect
STA	0	0	1	1	0	0	1	0	Store A direct
STH	0	0	1	1	0	1	0	0	Store A direct
SHLD	0	0	1	0	0	1	0	0	Store H & L direct
LHLD	0	0	1	0	1	0	1	0	Load H & L direct
XCHG	1	1	1	0	1	0	1	1	Exchange D & E, H & L Registers
STACK OPS									
PUSH B	1	1	0	0	0	1	0	1	Push register Pair B & C on stack
PUSH D	1	1	0	1	0	1	0	1	Push register Pair D & E on stack
PUSH H	1	1	1	0	0	1	0	1	Push register Pair H & L on stack
PUSH PSW	1	1	1	1	0	1	0	1	Push A and Flags on stack
POP B	1	1	0	0	0	0	0	1	Pop register Pair B & C off stack
POP D	1	1	0	1	0	0	0	1	Pop register Pair D & E off stack
POP H	1	1	1	0	0	0	0	1	Pop register Pair H & L off stack
POP PSW	1	1	1	1	0	0	0	1	Pop A and Flags off stack
XTHL	1	1	1	0	0	0	1	1	Exchange top of stack, H & L
SPHL	1	1	1	1	1	0	0	1	H & L to stack pointer
LXI SP	0	0	1	1	0	0	0	1	Load immediate stack pointer
INX SP	0	0	1	1	0	0	1	1	Increment stack pointer
DCX SP	0	0	1	1	1	0	1	1	Decrement stack pointer
JUMP									
JMP	1	1	0	0	0	0	1	1	Jump unconditional
JC	1	1	0	1	0	1	0	0	Jump on carry
JNC	1	1	0	1	0	0	1	0	Jump on no carry
JZ	1	1	0	0	1	0	1	0	Jump on zero
JNZ	1	1	0	0	0	1	0	1	Jump on no zero
JP	1	1	1	0	0	1	0	0	Jump on positive
JMP	1	1	1	1	0	1	0	0	Jump on minus
JPE	1	1	1	0	1	0	1	0	Jump on parity even
JPO	1	1	1	0	0	1	0	0	Jump on parity odd
PCHL	1	1	1	0	1	0	0	1	H & L to program counter
CALL									
CALL	1	1	0	0	1	1	0	1	Call unconditional
CC	1	1	0	1	1	1	0	0	Call on carry
CNC	1	1	0	1	0	1	0	0	Call on no carry
RETURN									
RET	1	1	0	0	1	0	0	1	Return
RC	1	1	0	1	1	0	0	0	Return on carry
RNC	1	1	0	1	0	0	0	0	Return on no carry
RZ	1	1	0	0	1	0	0	0	Return on zero
RNZ	1	1	0	0	0	0	0	0	Return on no zero
RP	1	1	1	1	0	0	0	0	Return on positive
RM	1	1	1	1	1	0	0	0	Return on minus
RPE	1	1	1	0	1	0	0	0	Return on parity even
RPO	1	1	1	0	0	0	0	0	Return on parity odd
RESTART									
RST	1	1	A	A	A	1	1	1	Restart
INPUT/OUTPUT									
IN	1	1	0	1	1	0	1	1	Input
OUT	1	1	0	1	0	0	1	1	Output
INCREMENT AND DECREMENT									
INR r	0	0	D	D	D	1	0	0	Increment register
DCR r	0	0	D	D	D	1	0	1	Decrement register
INR M	0	0	1	1	0	1	0	0	Increment memory
DCR M	0	0	1	1	0	1	0	1	Decrement memory
INX B	0	0	0	0	0	0	1	1	Increment B & C registers
INX D	0	0	0	1	0	0	1	1	Increment D & E registers
INX H	0	0	1	0	0	0	1	1	Increment H & L registers
DCX B	0	0	0	1	0	1	1	1	Decrement B & C
DCX D	0	0	0	1	1	0	1	1	Decrement D & E
DCX H	0	0	1	0	1	0	1	1	Decrement H & L
ADD									
ADD r	1	0	0	0	S	S	S	S	Add register to A
ADC r	1	0	0	0	1	S	S	S	Add register to A with carry
ADD M	1	0	C	0	0	1	1	0	Add memory to A
ADC M	1	0	C	0	1	1	1	0	Add memory to A with carry
ADI	1	1	0	0	0	1	1	0	Add immediate to A
ACI	1	1	0	0	1	1	1	0	Add immediate to A with carry
DAD B	0	0	0	0	1	0	0	1	Add B & C to H & L
DAD D	0	0	0	1	1	0	0	1	Add D & E to H & L
DAD H	0	0	1	0	1	0	0	1	Add H & L to H & L
DAD SP	0	0	1	1	1	0	0	1	Add stack pointer to H & L
SUBTRACT									
SUB r	1	0	0	1	0	S	S	S	Subtract register from A
SBB r	1	0	0	1	1	S	S	S	Subtract register from A with borrow
SUB M	1	0	0	1	0	1	1	0	Subtract memory from A
SBB M	1	0	0	1	1	1	1	0	Subtract memory from A with borrow
SUI	1	1	0	1	0	1	1	0	Subtract immediate from A
SBI	1	1	0	1	1	1	1	0	Subtract immediate from A with borrow

Table 6. Instruction Set Summary (Continued)

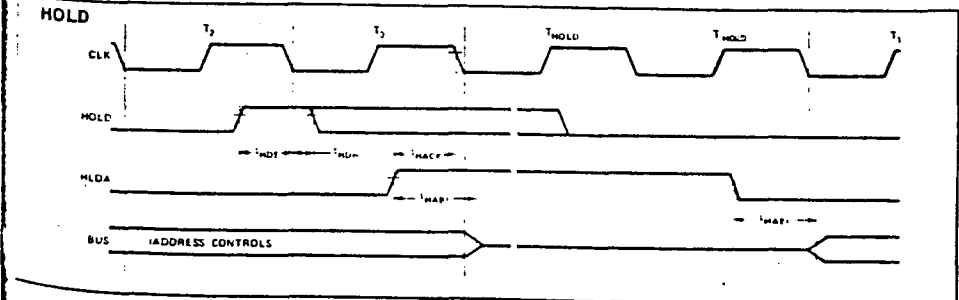
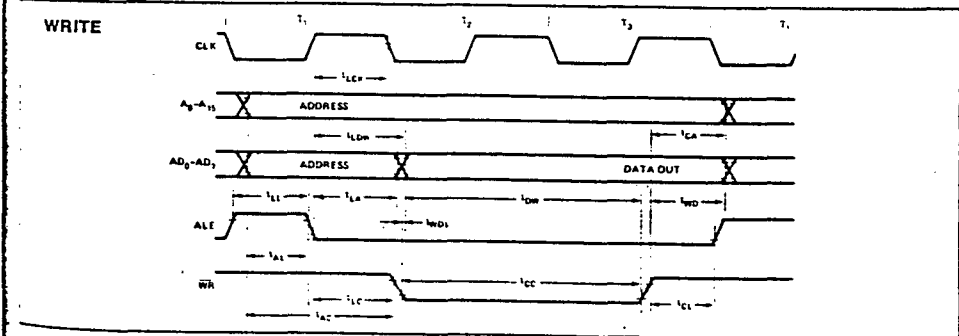
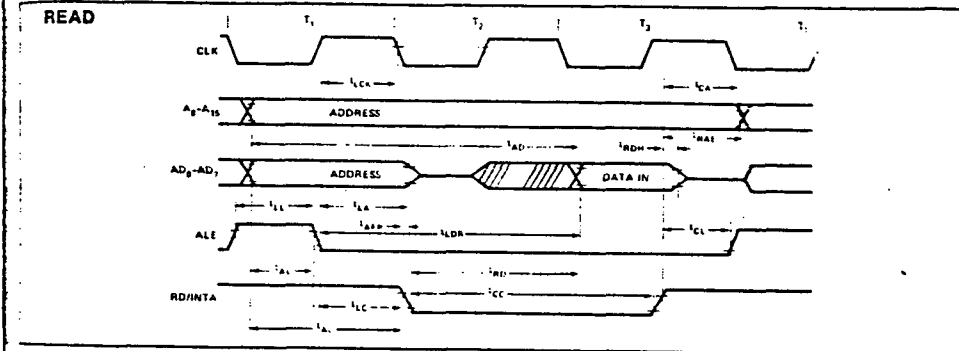
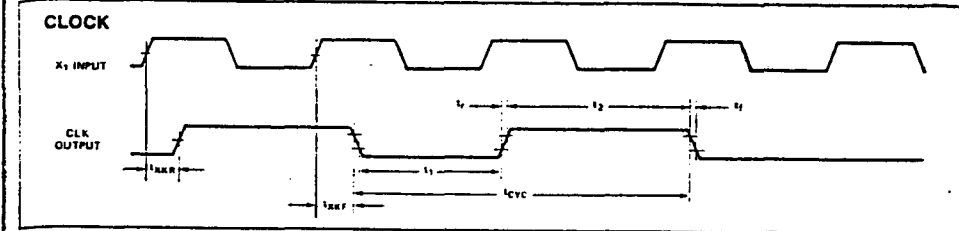
Mnemonic	D ₇	D ₆	D ₅	D ₄	D ₃	D ₂	D ₁	D ₀	Operations Description
LOGICAL									
ANA r	1	0	1	0	0	S	S	S	AND register with A
XRA r	1	0	1	0	1	S	S	S	Exclusive OR register with A
ORA r	1	0	1	1	0	S	S	S	OR register with A
CMP r	1	0	1	1	1	S	S	S	Compare register with A
ANA M	1	0	1	0	0	1	1	0	AND memory with A
XRA M	1	0	1	0	1	1	1	0	Exclusive OR memory with A
ORA M	1	0	1	1	0	1	1	0	OR memory with A
CMP M	1	0	1	1	1	1	1	0	Compare memory with A
ANI	1	1	1	0	0	1	1	0	AND immediate with A
XRI	1	1	1	0	1	1	1	0	Exclusive OR immediate with A
ORI	1	1	1	1	0	1	1	0	OR immediate with A
CPI	1	1	1	1	1	1	1	0	Compare immediate with A
ROTATE									
RLC	0	0	0	0	0	1	1	1	Rotate A left
RRC	0	0	0	0	1	1	1	1	Rotate A right
RAL	0	0	0	1	0	1	1	1	Rotate A left through carry
RAR	0	0	0	1	1	1	1	1	Rotate A right through carry

Mnemonic	D ₇	D ₆	D ₅	D ₄	D ₃	D ₂	D ₁	D ₀	Operations Description
SPECIALS									
CMA	0	0	1	0	1	1	1	1	Complement A
STC	0	0	1	1	0	1	1	1	Set carry
CMC	0	0	1	1	1	1	1	1	Complement carry
DAA	0	0	1	0	0	1	1	1	Decimal adjust A
CONTROL									
EI	1	1	1	1	1	0	1	1	Enable Interrupts
DI	1	1	1	1	0	0	1	1	Disable Interrupts
NOP	0	0	0	0	0	0	0	0	No-operation
HLT	0	1	1	1	0	1	1	0	Halt
NEW 8085A INSTRUCTIONS									
RIM	1	0	0	1	0	0	0	0	Read Interrupt Mask
SIM	1	0	0	1	1	0	0	0	Set Interrupt Mask

- NOTES:
 1. DDS or SSS: B 000, C 001, D 010, E 011, H 100, L 101, Memory 110, A 111.
 2. Two possible cycle times (6:12) indicate instruction cycles dependent on condition flags.

*All mnemonics copyrighted © Intel Corporation 1976.

WAVEFORMS



8085A/8085A-2

SINGLE CHIP 8-BIT N-CANNEL MICROPROCESSORS

- Single +5V Power Supply
- 100% Software Compatible with 8080A
- 1.3 μs Instruction Cycle (8085A); 0.8 μs (8085A-2)
- On-Chip Clock Generator (with External Crystal, LC or RC Network)
- On-Chip System Controller; Advanced Cycle Status Information Available for Large System Control
- Four Vectored Interrupt Inputs (One is Non-Maskable) Plus an 8080A-Compatible Interrupt
- Serial In/Serial Out Port
- Decimal, Binary and Double Precision Arithmetic
- Direct Addressing Capability to 64k Bytes of Memory

The Intel® 8085A is a complete 8 bit parallel Central Processing Unit (CPU). Its instruction set is 100% software compatible with the 8080A microprocessor, and it is designed to improve the present 8080A's performance by higher system speed. Its high level of system integration allows a minimum system of three IC's (8085A CPU, 8156 RAM/IO, and 8355/8755A (ROM/PROM/IO)) while maintaining total system expandability. The 8085A-2 is a faster version of the 8085A.

The 8085A incorporates all of the features that the 8224 (clock generator) and 8228 (system controller) provided for the 8080A, thereby offering a high level of system integration.

The 8085A uses a multiplexed data bus. The address is split between the 8 bit address bus and the 8 bit data bus. The on-chip address latches of 8155/8156/8355/8755A memory products allow a direct interface with the 8085A.

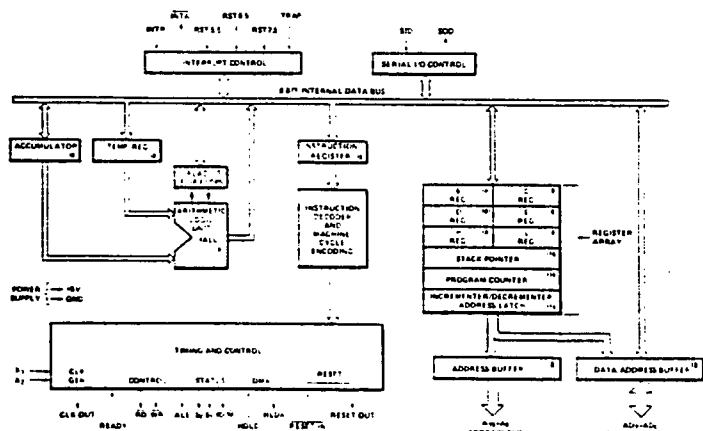


Figure 1. 8085A CPU Functional Block Diagram

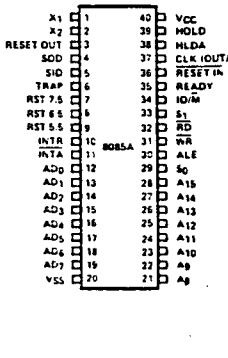


Figure 2. 8085A Pin Configuration

ABSOLUTE MAXIMUM RATINGS*

Storage Temperature Under Bias	0°C to 70°C
Operating Temperature	-65°C to +150°C
Voltage on Any Pin	-0.5V to +7V
Power Dissipation	1.5 Watt

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

D.C. CHARACTERISTICS (T_A = 0°C to 70°C, V_{CC} = 0V ±5%, V_{SS} = 0V; unless otherwise specified)

Symbol	Parameter	Min.	Max.	Units	Test Conditions
V _{IL}	Input Low Voltage	-0.5	+0.8	V	
V _{IH}	Input High Voltage	2.0	V _{CC} +0.5	V	
V _{OL}	Output Low Voltage		0.45	V	I _{OL} = 2mA
V _{OH}	Output High Voltage	2.4		V	I _{OH} = -400μA
I _{CC}	Power Supply Current		170	mA	
I _{IL}	Input Leakage		±10	μA	0 ≤ V _{IN} < V _{CC}
I _{OL}	Output Leakage		±10	μA	0.45V < V _{OUT} < V _{CC}
V _{LR}	Input Low Level RESET	-0.5	+0.8	V	
V _{HR}	Input High Level RESET	2.4	V _{CC} +0.5	V	
V _{HY}	Hysteresis, RESET	0.25		V	

A.C. CHARACTERISTICS ($T_A = 0^\circ\text{C}$ to 70°C , $V_{CC} = 0\text{V} \pm 5\%$, $V_{SS} = 0\text{V}$)

Symbol	Parameter	8085A ^[2]		8085A-2 ^[2]		Units
		Min.	Max.	Min.	Max.	
t_{CYC}	CLK Cycle Period	320	2000	200	2000	ns
t_1	CLK Low Time (Standard CLK Loading)	80		40		ns
t_2	CLK High Time (Standard CLK Loading)	120		70		ns
$t_{r,f}$	CLK Rise and Fall Time		30		30	ns
t_{ZKR}	X ₁ Rising to CLK Rising	30	120	30	100	ns
t_{XKF}	X ₁ Rising to CLK Falling	30	150	30	110	ns
t_{AC}	A_{8-15} Valid to Leading Edge of Control ^[1]	270		115		ns
t_{ACL}	A_{0-7} Valid to Leading Edge of Control	240		115		ns
t_{AD}	A_{0-15} Valid to Valid Data In		575		350	ns
t_{AFR}	Address Float After Leading Edge of READ (INTA)		0		0	ns
t_{AL}	A_{8-15} Valid Before Trailing Edge of ALE ^[1]	115		50		ns
t_{ALL}	A_{0-7} Valid Before Trailing Edge of ALE	90		50		ns
t_{ARY}	READY Valid from Address Valid		220		100	ns
t_{CA}	Address (A_{8-15}) Valid After Control	120		60		ns
t_{CC}	Width of Control Low (RD, WR, INTA) Edge of ALE	400		230		ns
t_{CL}	Trailing Edge of Control to Leading Edge of ALE	50		25		ns
t_{DW}	Data Valid to Trailing Edge of WRITE	420		230		ns
t_{HABE}	HLDA to Bus Enable		210		150	ns
t_{HABF}	Bus Float After HLDA		210		150	ns
t_{HACK}	HLDA Valid to Trailing Edge of CLK	110		40		ns
t_{HDL}	HOLD Hold Time	0		0		ns
t_{HDS}	HOLD Setup Time to Trailing Edge of CLK	170		120		ns
t_{INH}	INTR Hold Time	0		0		ns
t_{INS}	INTR, RST, and TRAP Setup Time to Falling Edge of CLK	160		150		ns
t_{IA}	Address Hold Time After ALE	100		50		ns
t_{LC}	Trailing Edge of ALE to Leading Edge of Control	130		60		ns
t_{LCK}	ALE Low During CLK High	100		50		ns
t_{LDR}	ALE to Valid Data During Read		460		270	ns
t_{LDW}	ALE to Valid Data During Write		200		120	ns
t_{LL}	ALE Width	140		80		ns
t_{LRY}	ALE to READY Stable		110		30	ns

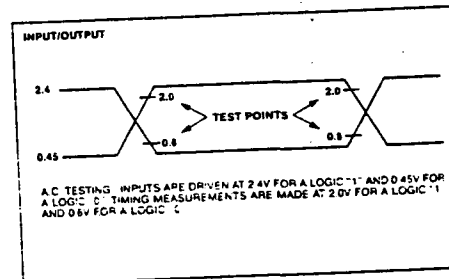
A.C. CHARACTERISTICS (Continued)

Symbol	Parameter	8085A ^[2]		8085A-2 ^[2]		Units
		Min.	Max.	Min.	Max.	
t_{RAE}	Trailing Edge of READ to Re-Enabling of Address	150		90		ns
t_{RD}	READ (or INTA) to Valid Data		300		150	ns
t_{RV}	Control Trailing Edge to Leading Edge of Next Control	400		220		ns
t_{RDH}	Data Hold Time After READ INTA ^[1]	0		0		ns
t_{RYH}	READY Hold Time	0		0		ns
t_{RYS}	READY Setup Time to Leading Edge of CLK	110		100		ns
t_{WD}	Data Valid After Trailing Edge of WRITE	100		60		ns
t_{WDL}	LEADING Edge of WRITE to Data Valid		40		20	ns

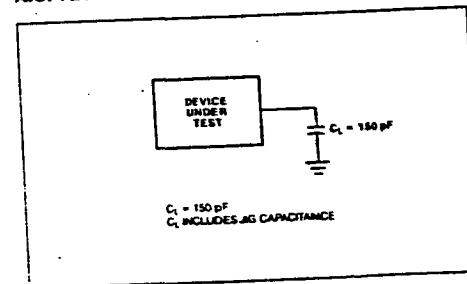
NOTES:

- A_{8-15} address Specs apply to IO/\bar{M} , S_0 , and S_1 except A_{8-15} are undefined during T_4 - T_6 of OF cycle whereas IO/\bar{M} , S_0 , and S_1 are stable.
- Test conditions:** $t_{CYC} = 320\text{ ns}$ (8085A)/200 ns (8085A-2); $C_L = 150\text{ pF}$.
- For all output timing where $C_L = 150\text{ pF}$ use the following correction factors:
 $25\text{ pF} < C_L < 150\text{ pF}$: -0.10 ns/pF
 $150\text{ pF} < C_L < 300\text{ pF}$: $+0.30\text{ ns/pF}$
- Output timings are measured with purely capacitive load.
- All timings are measured at output voltage $V_L = 0.8\text{V}$, $V_H = 2.0\text{V}$, and 1.5V with 20 ns rise and fall time on inputs.
- To calculate timing specifications at other values of t_{CYC} use Table 7.
- Data hold time is guaranteed under all loading conditions.

A.C. TESTING INPUT OUTPUT WAVEFORM



A.C. TESTING LOAD CIRCUIT



8155H/8156H/8155H-2/8156H-2 2048-BIT STATIC HMOS RAM WITH I/O PORTS AND TIMER

- Single +5V Power Supply with 10% Voltage Margins
- 30% Lower Power Consumption than the 8155 and 8156
- 100% Compatible with 8155 and 8156
- 256 Word x 8 Bits
- Completely Static Operation
- Internal Address Latch
- 2 Programmable 8-Bit I/O Ports
- 1 Programmable 6-Bit I/O Port
- Programmable 14-Bit Binary Counter/Timer
- Compatible with 8085AH, 8085A and 8088 CPU
- Multiplexed Address and Data Bus
- Available in EXPRESS
 - Standard Temperature Range
 - Extended Temperature Range

The Intel[®] 8155H and 8156H are RAM and I/O chips implemented in N-Channel, depletion load, silicon gate technology (HMOS), to be used in the 8085AH and 8088 microprocessor systems. The RAM portion is designed with 2048 static cells organized as 256 x 8. They have a maximum access time of 400 ns to permit use with no wait states in 8085AH CPU. The 8155H-2 and 8156H-2 have maximum access times of 330 ns for use with the 8085AH-2 and the 5 MHz 8088 CPU.

The I/O portion consists of three general purpose I/O ports. One of the three ports can be programmed to be status pins, thus allowing the other two ports to operate in handshake mode.

A 14-bit programmable counter/timer is also included on chip to provide either a square wave or terminal count pulse for the CPU system depending on timer mode.

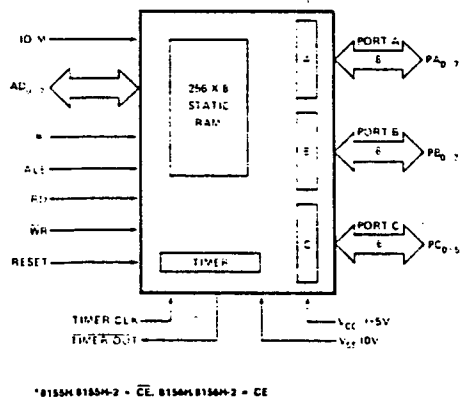


Figure 1. Block Diagram

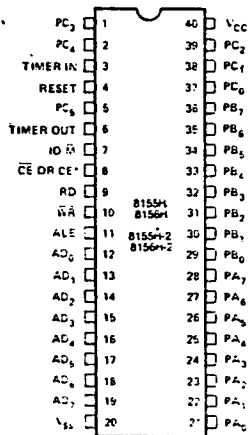


Figure 2. Pin Configuration

Table 1. Pin Description

Symbol	Type	Name and Function
RESET	I	Reset: Pulse provided by the 8085AH to initialize the system (connect to 8085AH RESET OUT). Input high on this line resets the chip and initializes the three I/O ports to input mode. The width of RESET pulse should typically be two 8085AH clock cycle times.
AD ₀₋₇	I/O	Address/Data: 8-state Address-Data lines that interface with the CPU lower 8-bit Address/Data Bus. The 8-bit address is latched into the address latch inside the 8155H/56H on the falling edge of ALE. The 8-bit data is either for the memory section or the I/O section depending on the IO M input. The 8-bit data is either written into the chip or read from the chip, depending on the WR or RD input signal.
CE or CE	I	Chip Enable: On the 8155H, this pin is \overline{CE} and is ACTIVE LOW. On the 8156H, this pin is CE and is ACTIVE HIGH.
RD	I	Read Control: Input low on this line with the Chip Enable active enables and AD ₀₋₇ buffers. If IO M pin is low, the RAM content will be read out to the AD bus. Otherwise the content of the selected I/O port or command/status registers will be read to the AD bus.
WR	I	Write Control: Input low on this line with the Chip Enable active causes the data on the Address-Data bus to be written to the RAM or I/O ports and command/status register, depending on IO M.
ALE	I	Address Latch Enable: This control signal latches both the address on the AD ₀₋₇ lines and the state of the Chip Enable and IO M into the chip at the falling edge of ALE.
IO M	I	I/O Memory: Selects memory if low and I/O and command status registers if high.
PC ₀₋₇ (8)	LO	Port A: These 8 pins are general purpose I/O pins. The in out direction is selected by programming the command register.
PB ₀₋₇ (8)	I/O	Port B: These 8 pins are general purpose I/O pins. The in out direction is selected by programming the command register.
PC ₀₋₅ (6)	I/O	Port C: These 6 pins can function as either input port, output port, or as control signals for PA and PB. Programming is done through the command register. When PC ₀₋₅ are used as control signals, they will provide the following: PC ₀ — A INTR (Port A Interrupt) PC ₁ — ABF (Port A Buffer Full) PC ₂ — A STB (Port A Strobe) PC ₃ — B INTR (Port B Interrupt) PC ₄ — B BF (Port B Buffer Full) PC ₅ — B STB (Port B Strobe)
TIMER IN	I	Timer Input: Input to the counter-timer.
TIMER OUT	O	Timer Output: This output can be either a square wave or a pulse, depending on the timer mode.
Vcc		Voltage: +5 volt supply.
Vss		Ground: Ground reference.

FUNCTIONAL DESCRIPTION

The 8155H/8156H contains the following:

- 2K Bit Static RAM organized as 256 x 8
- Two 8-bit I/O ports PA & PB and one 6-bit I/O port PC
- 14-bit timer-counter

The IO M I/O/Memory Select pin selects either the five registers Command, Status, PA₀₋₇, PB₀₋₇, PC₀₋₅ or the memory RAM portion.

The 8-bit address on the Address-Data lines, Chip Enable (CE or \overline{CE}), and IO M are all latched on-chip at the falling edge of ALE.

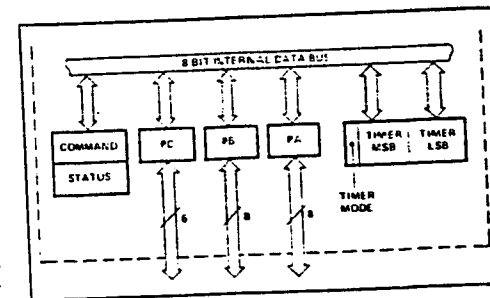


Figure 3. 8155H/8156H Internal Registers

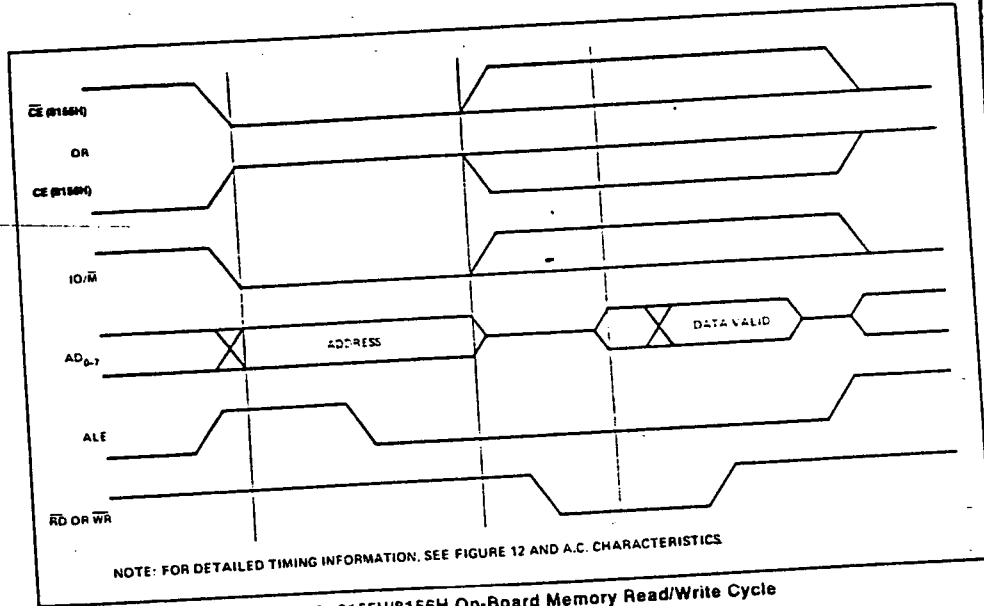


Figure 4. 8155H/8156H On-Board Memory Read/Write Cycle

PROGRAMMING OF THE COMMAND REGISTER

The command register consists of eight latches. Four bits .0-3 define the mode of the ports, two bits .4-5 enable or disable the interrupt from port C when it acts as control port, and the last two bits 6-7 are for the timer.

The command register contents can be altered at any time by using the I/O address XXXXX000 during a WRITE operation with the Chip Enable active and IO/M = 1. The meaning of each bit of the command register may never be read.

READING THE STATUS REGISTER

The status register consists of seven latches, one for each bit: six .0-5 for the status of the ports and one .6 for the status of the timer.

The status of the timer and the I/O section can be polled by reading the Status Register (Address XXXXX000). Status word format is shown in Figure 6. Note that you may never write to the status register since the command register shares the same I/O address and the command register is selected when a write to that address is issued

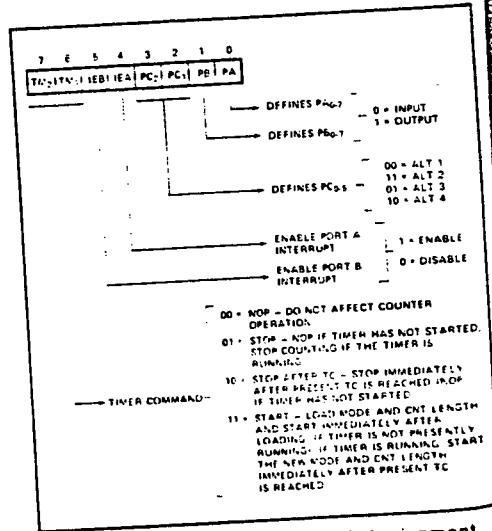


Figure 5. Command Register Bit Assignment

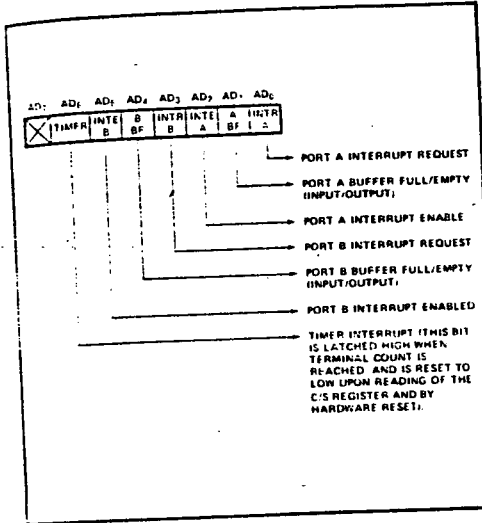


Figure 6. Status Register Bit Assignment

INPUT/OUTPUT SECTION

The I/O section of the 8155H/8156H consists of five registers: (See Figure 7.)

- **Command/Status Register (C/S)** — Both registers are assigned the address XXXXX000. The C/S address serves the dual purpose.

When the C/S registers are selected during WRITE operation, a command is written into the command register. The contents of this register are not accessible through the pins.

When the C/S address XXXXX000 is selected during a READ operation, the status information of the I/O ports and the timer becomes available on the AD₀₋₇ lines.

- **PA Register** — This register can be programmed to be either input or output ports depending on the status of the contents of the C/S Register. Also depending on the command, this port can operate in either the basic mode or the strobed mode. See timing diagram. The I/O pins assigned in relation to this register are PA₀₋₇. The address of this register is XXXXX010.
- **PB Register** — This register functions the same as PA Register. The I/O pins assigned are PB₀₋₇. The address of this register is XXXXX011.
- **PC Register** — This register has the address XXXXX012 and contains only 6 bits. The 6 bits can be programmed to be either input ports, output ports or as control signals for PA and PB by properly programming the AD₂ and AD₃ bits of the C/S register.

When PC₀₋₅ is used as a control port, 3 bits are assigned for Port A and 3 for Port B. The first bit is an

interrupt that the 8155H sends out. The second is an output signal indicating whether the buffer is full or empty, and the third is an input pin to accept a strobe for the strobed input mode. (See Table 2.)

When the 'C' port is programmed to either ALT3 or ALT4, the control signals for PA and PB are initialized as follows:

CONTROL	INPUT MODE	OUTPUT MODE
BF	Low	Low
INTR	Low	High
STB	Input Control	Input Control

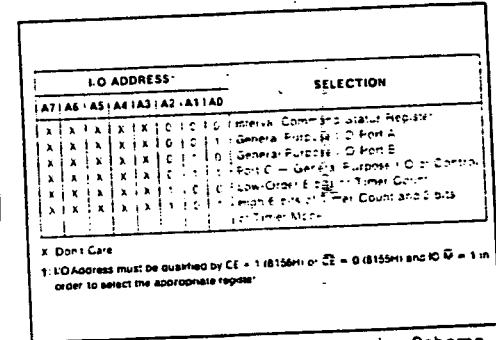


Figure 7. I/O Port and Timer Addressing Scheme

Figure 8 shows how I/O PORTS A and B are structured within the 8155H and 8156H:

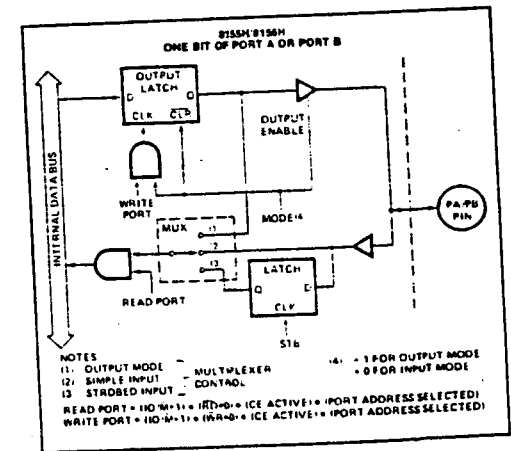


Figure 8. 8155H/8156H Port Functions

Table 2. Port Control Assignment

Pin	ALT 1	ALT 2	ALT 3	ALT 4
PC0	Input Port	Output Port	A INTR - Port A Interrupt	A INTR - Port A Interrupt
PC1	Input Port	Output Port	A BF - Port A Buffer Full	A BF - Port A Buffer Full
PC2	Input Port	Output Port	A STB - Port A Strobe	A STB - Port A Strobe
PC3	Input Port	Output Port	Output Port	B INTR - Port B Interrupt
PC4	Input Port	Output Port	Output Port	B BF - Port B Buffer Full
PC5	Input Port	Output Port	Output Port	B STB - Port B Strobe

Note in the diagram that when the I/O ports are programmed to be output ports, the contents of the output ports can still be read by a READ operation when appropriately addressed.

The outputs of the 8155H/8156H are "glitch-free" meaning that you can write a "1" to a bit position that was previously "1" and the level at the output pin will not change.

Note also that the output latch is cleared when the port enters the input mode. The output latch cannot be loaded by writing to the port if the port is in the input mode. The result is that each time a port mode is changed from input to output, the output pins will go low. When the 8155H/56H is RESET, the output latches are all cleared and all 3 ports enter the input mode.

When in the ALT 1 or ALT 2 modes, the bits of PORT C are structured like the diagram above in the simple input or output mode, respectively.

Reading from an input port with nothing connected to the pins will provide unpredictable results.

Figure 9 shows how the 8155H/8156H I/O ports might be configured in a typical MCS-85 system.

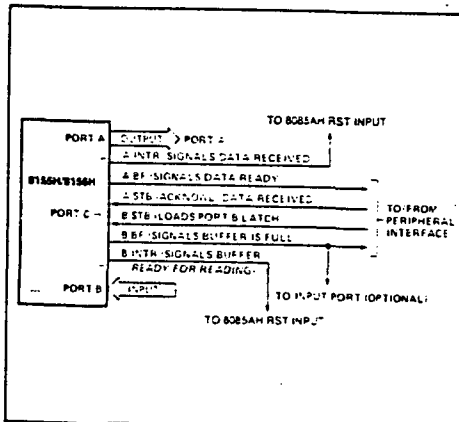


Figure 9. Example: Command Register = 00111001

TIMER SECTION

The timer is a 14-bit down-counter that counts the TIMER IN pulses and provides either a square wave or pulse when terminal count (TC) is reached.

The timer has the I/O address XXXXX100 for the low order byte of the register and the I/O address XXXXX101 for the high order byte of the register. (See Figure 7.)

To program the timer, the COUNT LENGTH REG is loaded first, one byte at a time, by selecting the timer addresses. Bits 0-13 of the high order count register will specify the length of the next count and bits 14-15 of the high order register will specify the timer output mode (see Figure 10). The value loaded into the count length register can have any value from 2H through 3FFH in Bits 0-13.

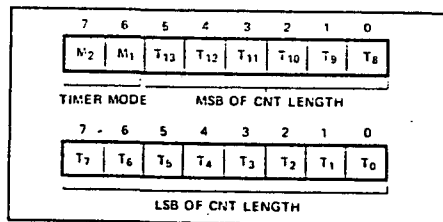


Figure 10. Timer Format

There are four modes to choose from: M2 and M1 define the timer mode, as shown in Figure 11.

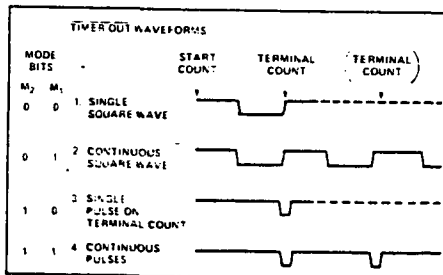


Figure 11. Timer Modes

Bits 6-7 (TM2 and TM1) of command register contents are used to start and stop the counter. There are four commands to choose from:

TM2	TM1	
0	0	NOP — Do not affect counter operation.
0	1	STOP — NOP if timer has not started; stop counting if the timer is running.
1	0	STOP AFTER TC — Stop immediately after present TC is reached (NOP if timer has not started).
1	1	START — Load mode and CNT length and start immediately after loading (if timer is not presently running). If timer is running, start the new mode and CNT length immediately after present TC is reached.

Note that while the counter is counting, you may load a new count and mode into the count length registers. Before the new count and mode will be used by the counter, you must issue a START command to the counter. This applies even though you may only want to change the count and use the previous mode.

In case of an odd-numbered count, the first half-cycle of the squarewave output, which is high, is one count longer than the second, low, half-cycle, as shown in Figure 12.

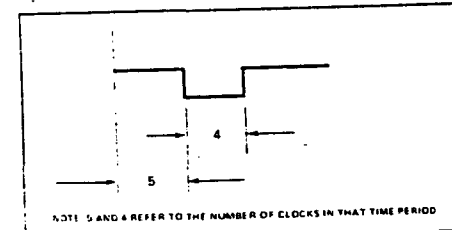


Figure 12. Asymmetrical Square-Wave Output Resulting from Count of 9

The counter in the 8155H is not initialized to any particular mode or count when hardware RESET occurs, but RESET does stop the counting. Therefore, counting cannot begin following RESET until a START command is issued via the C/S register.

Please note that the timer circuit on the 8155H/8156H chip is designed to be a square-wave timer, not an event counter. To achieve this, it counts down by twos twice in completing one cycle. Thus, as registers do not contain values directly representing the number of TIMER IN pulses received. You cannot load an initial value of 1 into the count register and cause the timer to operate, as its terminal count value is 10 (binary) or 2 (decimal). (For the detection of single pulses, it is suggested that one of the hardware interrupt pins on the 8085AH be used.) After the timer has started counting down, the values residing in the count registers can be used to calculate the actual number of TIMER IN pulses required to complete the timer cycle if desired. To obtain the remaining count, perform the following operations in order:

1. Stop the count
2. Read in the 16-bit value from the count length registers
3. Reset the upper two mode bits
4. Reset the carry and rotate right one position all 16 bits through carry
5. If carry is set, add 1/2 of the full original count (1/2 full count — 1 if full count is odd).

Note: If you started with an odd count and you read the count length register before the third count pulse occurs, you will not be able to discern whether one or two counts has occurred. Regardless of this, the 8155H/56H always counts out the right number of pulses in generating the TIMER OUT waveforms.

8085A MINIMUM SYSTEM CONFIGURATION

Figure 13a shows a minimum system using three chips, containing:

- 256 Bytes RAM
- 2K Bytes ROM
- 38 I/O Pins
- 1 Interval Timer
- 4 Interrupt Levels

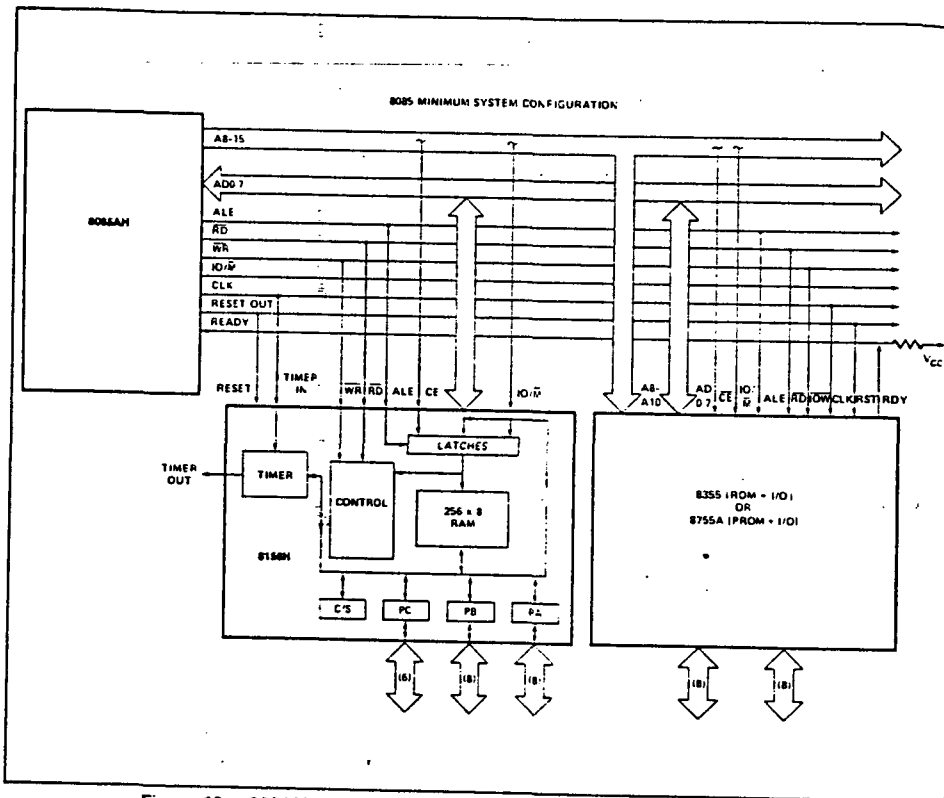


Figure 13a. 8085AH Minimum System Configuration (Memory Mapped I/O)

8088 FIVE CHIP SYSTEM

Figure 13b shows a five chip system containing:

- 1.25K Bytes RAM
- 2K Bytes ROM

- 38 I/O Pins
- 1 Interval Timer
- 2 Interrupt Levels

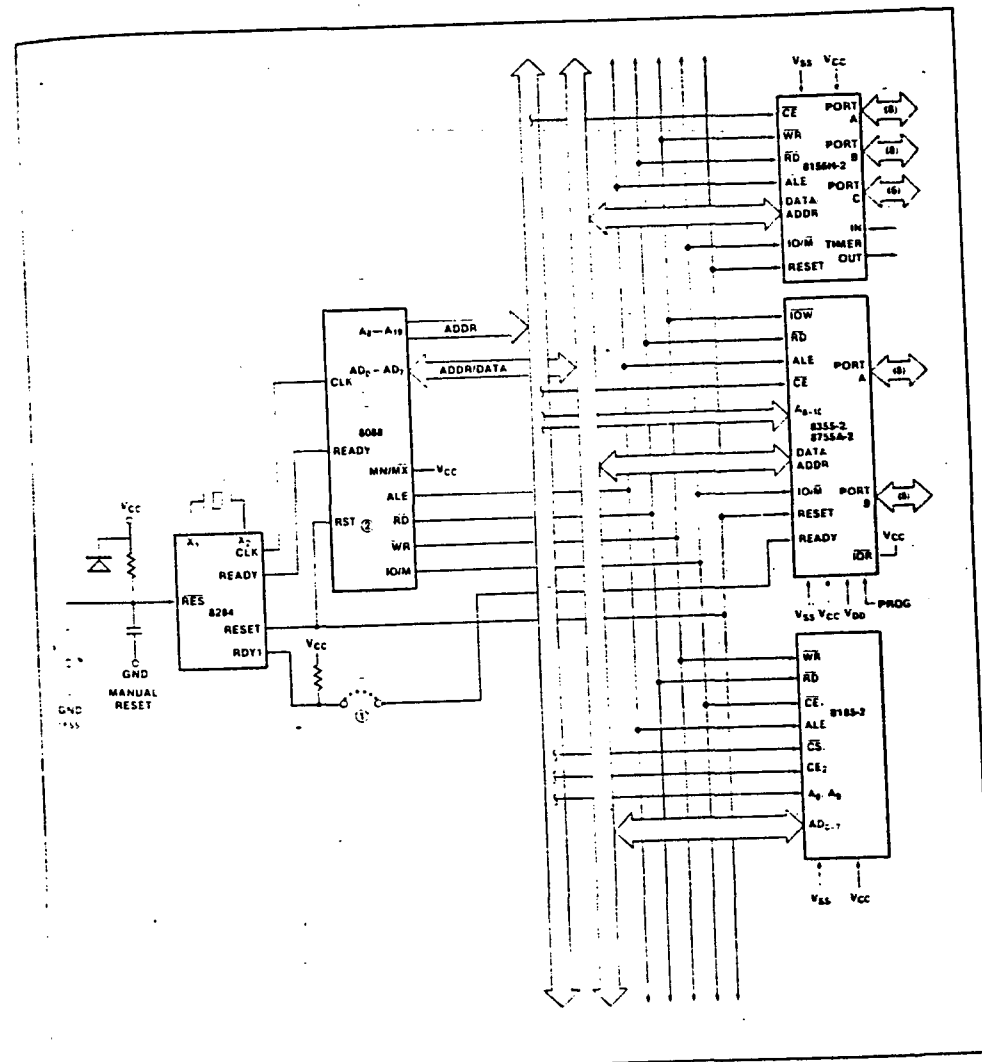


Figure 13b. 8088 Five Chip System Configuration

ABSOLUTE MAXIMUM RATINGS*

Temperature Under Bias 0°C to +70°C
 Storage Temperature -65°C to +150°C
 Voltage on Any Pin
 With Respect to Ground -0.5V to +7V
 Power Dissipation 1.5W

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

D.C. CHARACTERISTICS (T_A = 0°C to 70°C, V_{CC} = 5V ± 10%)

Symbol	Parameter	Min.	Max.	Units	Test Conditions
V _{IL}	Input Low Voltage	-0.5	0.8	V	
V _{IH}	Input High Voltage	2.0	V _{CC} +0.5	V	
V _{OL}	Output Low Voltage		0.45	V	I _{OL} = 2mA
V _{OH}	Output High Voltage	2.4		V	I _{OH} = -400µA
I _{IL}	Input Leakage		±10	µA	0V ≤ V _{IN} ≤ V _{CC}
I _{LO}	Output Leakage Current		±10	µA	0.45V ≤ V _{OUT} ≤ V _{CC}
I _{CC}	V _{CC} Supply Current		125	mA	
I _{IL} (CE)	Chip Enable Leakage 8155H 8156H		+100 -100	µA	0V ≤ V _{IN} ≤ V _{CC}

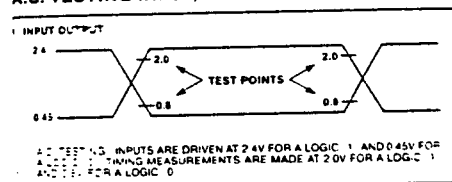
A.C. CHARACTERISTICS (T_A = 0°C to 70°C, V_{CC} = 5V ± 10%)

Symbol	Parameter	8155H/8156H		8155H-2/8156H-2		Units														
		Min.	Max.	Min.	Max.															
t _{AL}	Address to Latch Set Up Time	50		30		ns														
t _{LA}	Address Hold Time after Latch	80		30		ns														
t _{LC}	Latch to READ/WRITE Control	100		40		ns														
t _{RD}	Valid Data Out Delay from READ Control		170		140	ns														
t _{AD}	Address Stable to Data Out Valid		400		330	ns														
t _{LL}	Latch Enable Width	100		70		ns														
t _{RDF}	Data Bus Float After READ	0	100	0	80	ns														
t _{CL}	READ/WRITE Control to Latch Enable	20		10		ns														
t _{CC}	READ/WRITE Control Width	250		200		ns														
t _{DW}	Data In to WRITE Set Up Time	150		100		ns														
t _{WD}	Data In Hold Time After WRITE	25		25		ns														
t _{RV}	Recovery Time Between Controls	300		200		ns														
t _{WP}	WRITE to Port Output		400		300	ns														
t _{PR}	Port Input Setup Time	70		50		ns														
t _{RP}	Port Input Hold Time	50		10		ns														
t _{SBF}	Strobe to Buffer Full		400		300	ns														
t _{SS}	Strobe Width	200		150		ns </tr <tr> <td>t_{RBE}</td> <td>READ to Buffer Empty</td> <td></td> <td>400</td> <td></td> <td>300</td> <td>ns</td> </tr> <tr> <td>t_{SI}</td> <td>Strobe to INTR On</td> <td></td> <td>400</td> <td></td> <td>300</td> <td>ns</td> </tr>	t _{RBE}	READ to Buffer Empty		400		300	ns	t _{SI}	Strobe to INTR On		400		300	ns
t _{RBE}	READ to Buffer Empty		400		300	ns														
t _{SI}	Strobe to INTR On		400		300	ns														

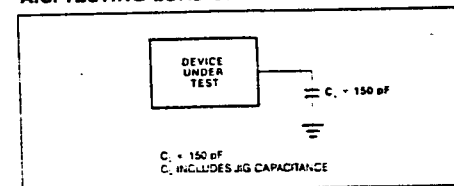
A.C. CHARACTERISTICS (Continued) (T_A = 0°C to 70°C, V_{CC} = 5V ± 10%)

Symbol	Parameter	8155H/8156H		8155H-2/8156H-2		Units
		Min.	Max.	Min.	Max.	
t _{RD}	READ to INTR Off		400		300	ns
t _{PSS}	Port Setup Time to Strobe Strobe	50		0		ns
t _{PHS}	Port Hold Time After Strobe	120		100		ns
t _{SBE}	Strobe to Buffer Empty		400		300	ns
t _{WBF}	WRITE to Buffer Full		400		300	ns
t _{WI}	WRITE to INTR Off		400		300	ns
t _{T-LOW}	TIMER-IN to TIMER-OUT Low		400		300	ns
t _{T-HIGH}	TIMER-IN to TIMER-OUT High		400		300	ns
t _{RE}	Data Bus Enable from READ Control	10		10		ns
t ₁	TIMER-IN Low Time	80		40		ns
t ₂	TIMER-IN High Time	120		70		ns

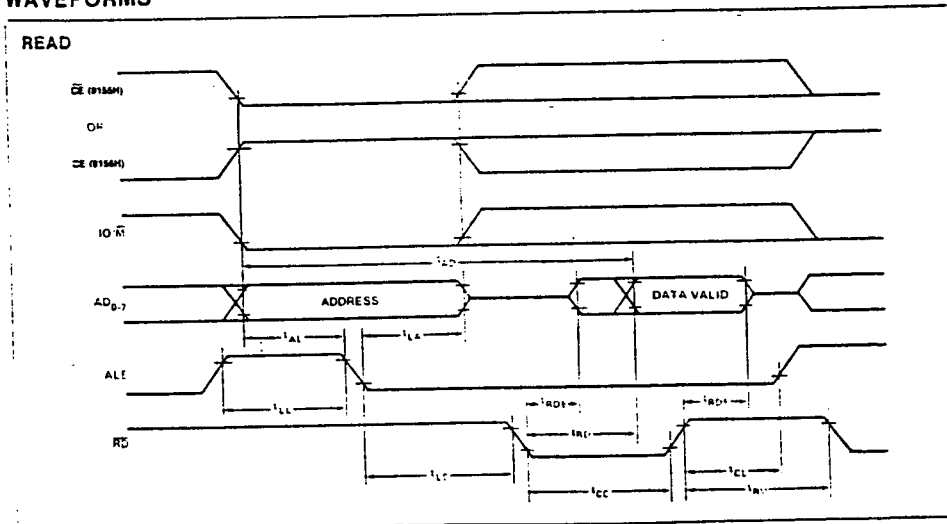
A.C. TESTING INPUT, OUTPUT WAVEFORM



A.C. TESTING LOAD CIRCUIT



WAVEFORMS





Operational Amplifiers/Buffers

LM124/LM224/LM324, LM124A/LM224A/LM324A, LM2902 Low Power Quad Operational Amplifiers

General Description

The LM124 series consists of four independent, high gain, internally frequency compensated operational amplifiers which were designed specifically to operate from a single power supply over a wide range of voltages. Operation from split power supplies is also possible and the low power supply current drain is independent of the magnitude of the power supply voltage.

Application areas include transducer amplifiers, dc gain blocks and all the conventional op amp circuits which now can be more easily implemented in single power supply systems. For example, the LM124 series can be directly operated off of the standard +5 V_{DC} power supply voltage which is used in digital systems and will easily provide the required interface electronics without requiring the additional ±15 V_{DC} power supplies.

Unique Characteristics

- In the linear mode the input common-mode voltage range includes ground and the output voltage can also swing to ground, even though operated from only a single power supply voltage.
- The unity gain cross frequency is temperature compensated.
- The input bias current is also temperature compensated.

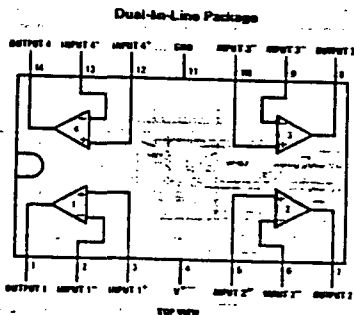
Advantages

- Eliminates need for dual supplies
- Four internally compensated op amps in a single package
- Allows directly sensing near GND and V_{OUT} also goes to GND
- Compatible with all forms of logic
- Power drain suitable for battery operation

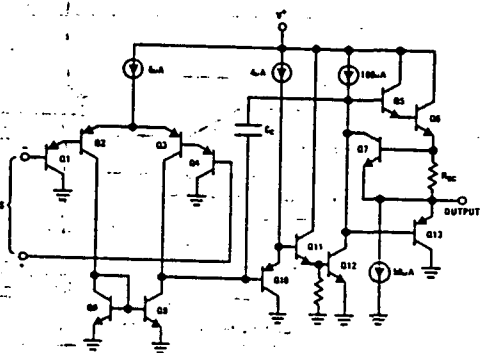
Features

- Internally frequency compensated for unity gain
- Large dc voltage gain 100 dB
- Wide bandwidth (unity gain) 1 MHz (temperature compensated)
- Wide power supply range:
 - Single supply 3 V_{DC} to 30 V_{DC}
 - or dual supplies ±1.5 V_{DC} to ±15 V_{DC}
- Very low supply current drain (800 μA) — essentially independent of supply voltage (1 mW/op amp at +5 V_{DC})
- Low input biasing current (temperature compensated) 45 nA_{DC}
- Low input offset voltage and offset current 2 mV_{DC} 5 nA_{DC}
- Input common-mode voltage range includes ground
- Differential input voltage range equal to the power supply voltage
- Large output voltage swing 0 V_{DC} to V* - 1.5 V_{DC}

Connection Diagram



Schematic Diagram (Each Amplifier)



Order Number LM124J, LM124AJ,
LM224J, LM224AJ, LM324J,
LM324AJ or LM2902J
See NS Package J14A

Order Number LM324N, LM324AN
or LM2902N
See NS Package N14A

Absolute Maximum Ratings

- Supply Voltage, V* 30 VDC
- Differential Input Voltage 32 VDC
- Input Voltage -0.3 VDC to +26 VDC
- Power Dissipation (Note 1) 570 mW
- Molded DIP 900 mW
- Cavity DIP 800 mW
- Continuous
- Output Short-Circuit to GND (One Amplifier) (Note 2) 570 mW
- V* ≤ 15 VDC and T_A = 25°C

LM124/LM224/LM324
LM124A/LM224A/LM324A

LM2902

26 VDC or ±13 VDC
26 VDC
-0.3 VDC to +26 VDC

570 mW
900 mW
800 mW
Continuous

Input Current (V_{IN} < -0.3 VDC) (Note 3)
Operating Temperature Range
LM324/LM324A
LM224/LM224A
LM124/LM124A
Storage Temperature Range
Lead Temperature (Soldering, 10 seconds)

LM124/LM224/LM324
LM124A/LM224A/LM324A

50 mA
-40°C to +85°C
50 mA
0°C to +70°C
-25°C to +85°C
-55°C to +125°C
-65°C to +150°C
300°C
-85°C to +180°C
300°C

Electrical Characteristics (V* = +6.0 VDC, Note 4)

PARAMETER	CONDITIONS	LM124A	LM224A	LM324A	LM124/LM224	LM324	LM2902	UNITS
Input Offset Voltage	T _A = 25°C, (Note 5)	MIN 1	MIN 1	MIN 1	MIN 1	MIN 1	MIN 1	mVDC
Input Offset Current	I _{IN(1)} or I _{IN(-1)} , T _A = 25°C	20	40	45	12	17	12	nADC
Input Bias Current	I _{IN(1)} - I _{IN(-1)} , T _A = 25°C	2	3	3	3	3	3	nADC
Input Common-Mode Voltage Range (Note 7)	V* = 30 VDC, T _A = 25°C	2	2	2	2	2	2	VDC
Supply Current	V* = 30 VDC, T _A = 25°C	1.5	1.5	1.5	1.5	1.5	1.5	mADC
Large Signal Voltage Gain	R _L = ∞, V _{CC} = 30V, (LM2902 V _{CC} = 26V) R _L ≥ 2 kΩ, T _A = 25°C	0.7	0.7	0.7	0.7	0.7	0.7	mADC
Output Voltage Swing	V* = 15 VDC (For Large V _O Swing) R _L ≥ 2 kΩ, T _A = 25°C	50	50	50	50	50	50	V/mV
Common-Mode Rejection Ratio	R _L = 2 kΩ, T _A = 25°C, (LM2902 R _L ≥ 10 kΩ)	70	70	70	70	70	70	dB
Power Supply Rejection Ratio	DC, T _A = 25°C	65	65	65	65	65	65	dB
Amplifier-to-Amplifier Coupling (Note 8)	f = 1 kHz to 20 kHz, T _A = 25°C (Input Referenced)	-120	-120	-120	-120	-120	-120	dB
Output Current Source	V _{IN} = 1 VDC, V _{IN} = 0 VDC V* = 15 VDC, T _A = 25°C	20	20	20	20	20	20	mADC
Sink	V _{IN} = 1 VDC, V _{IN} = 0 VDC V* = 15 VDC, T _A = 25°C	10	10	10	10	10	10	mADC
Short Circuit to Ground	V _{IN} = 1 VDC, V _{IN} = 0 VDC T _A = 25°C, V _O = 200 mVDC	12	12	12	12	12	12	mADC

Electrical Characteristics (Continued)

PARAMETER	CONDITIONS	LM124A			LM224A			LM324A			LM124/LM224			LM324			LM2902			UNITS		
		MIN	TYP	MAX	MIN	TYP	MAX	MIN	TYP	MAX	MIN	TYP	MAX	MIN	TYP	MAX	MIN	TYP	MAX			
Input Offset Voltage	(Note 5)			4			4			8			27			18			±10	mVDC		
Input Offset Voltage Drift	$R_S = 0\Omega$			7			7			7			7			7			7	$\mu\text{V}/^\circ\text{C}$		
Input Offset Current	$I_{IN(+)} - I_{IN(-)}$			30			30			75			4100			±150			45	±200	nADC	
Input Offset Current Drift				10			10			10			10			10			10	$\mu\text{ADC}/^\circ\text{C}$		
Input Bias Current	$I_{IN(+)} \text{ or } I_{IN(-)}$			40			40			200			40			300			40	500	nADC	
Input Common Mode Voltage Range (Note 7)	$V^+ = 30\text{VDC}$	0		$V^+ - 2$	0		$V^+ - 2$	0		$V^+ - 2$	0		$V^+ - 2$	0		$V^+ - 2$	0		$V^+ - 2$	0	$V^+ - 2$	VDC
Large Signal Voltage Gain	$V^+ = +15\text{VDC}$ (For Large V_O Swing) $R_L \geq 2\text{k}\Omega$	25			25			15			25			15					15		V/mV	
Output Voltage Swing	$V^+ = +30\text{VDC}$, $R_L = 2\text{k}\Omega$ $R_L \geq 10\text{k}\Omega$	V_{OH}	26		26		26			26			26			26			22		VDC	
		V_{OL}	27	28	27	28	27	28	27	28	27	28	27	28	27	28	27	28	23	24		VDC
Output Current	$V^+ = +5\text{VDC}$, $R_L \leq 10\text{k}\Omega$	Source	5		5		5			5			5			5			5		mADC	
		Sink	20		20		20			20			20			20			20		20	mADC
Differential Input Voltage	(Note 7)			32			32			32			32			32			26		VDC	

Note 1: For operating at high temperatures, the LM324/LM324A, LM2902 must be derated based on a $+125^\circ\text{C}$ maximum junction temperature and a thermal resistance of $175^\circ\text{C}/\text{W}$ which applies for the device soldered in a printed circuit board, operating in a still air ambient. The LM224/LM224A and LM124/LM124A can be derated based on a $+150^\circ\text{C}$ maximum junction temperature. The dissipation is the total of all four amplifiers—use external resistors, where possible, to allow the amplifier to saturate or to reduce the power which is dissipated in the integrated circuit.

Note 2: Short circuits from the output to V^+ can cause excessive heating and eventual destruction. The maximum output current is approximately 40 mA independent of the magnitude of V^+ . At values of supply voltage in excess of $+15\text{VDC}$, continuous short-circuits can exceed the power dissipation ratings and cause eventual destruction. Destructive dissipation can result from simultaneous shorts on all amplifiers.

Note 3: This input current will only exist when the voltage at any of the input leads is driven negative. It is due to the collector-base junction of the input PNP transistors becoming forward biased and thereby acting as input diode clamps. In addition to this diode action, there is also lateral NPN parasitic transistor action on the IC chip. This transistor action can cause the output voltages of the op amps to go to the V^+ voltage level (or to ground for a large overdrive) for the time duration that an input is driven negative. This is not destructive and normal output states will re-establish when the input voltage, which was negative, again returns to a value greater than -0.3VDC (at 25°C).

Note 4: These specifications apply for $V^+ = +5\text{VDC}$ and $-55^\circ\text{C} \leq T_A \leq +125^\circ\text{C}$, unless otherwise stated. With the LM224/LM224A, all temperature specifications are limited to $-25^\circ\text{C} \leq T_A \leq +85^\circ\text{C}$, the LM324/LM324A temperature specifications are limited to $0^\circ\text{C} \leq T_A \leq +70^\circ\text{C}$, and the LM2902 specifications are limited to $-40^\circ\text{C} \leq T_A \leq +85^\circ\text{C}$.

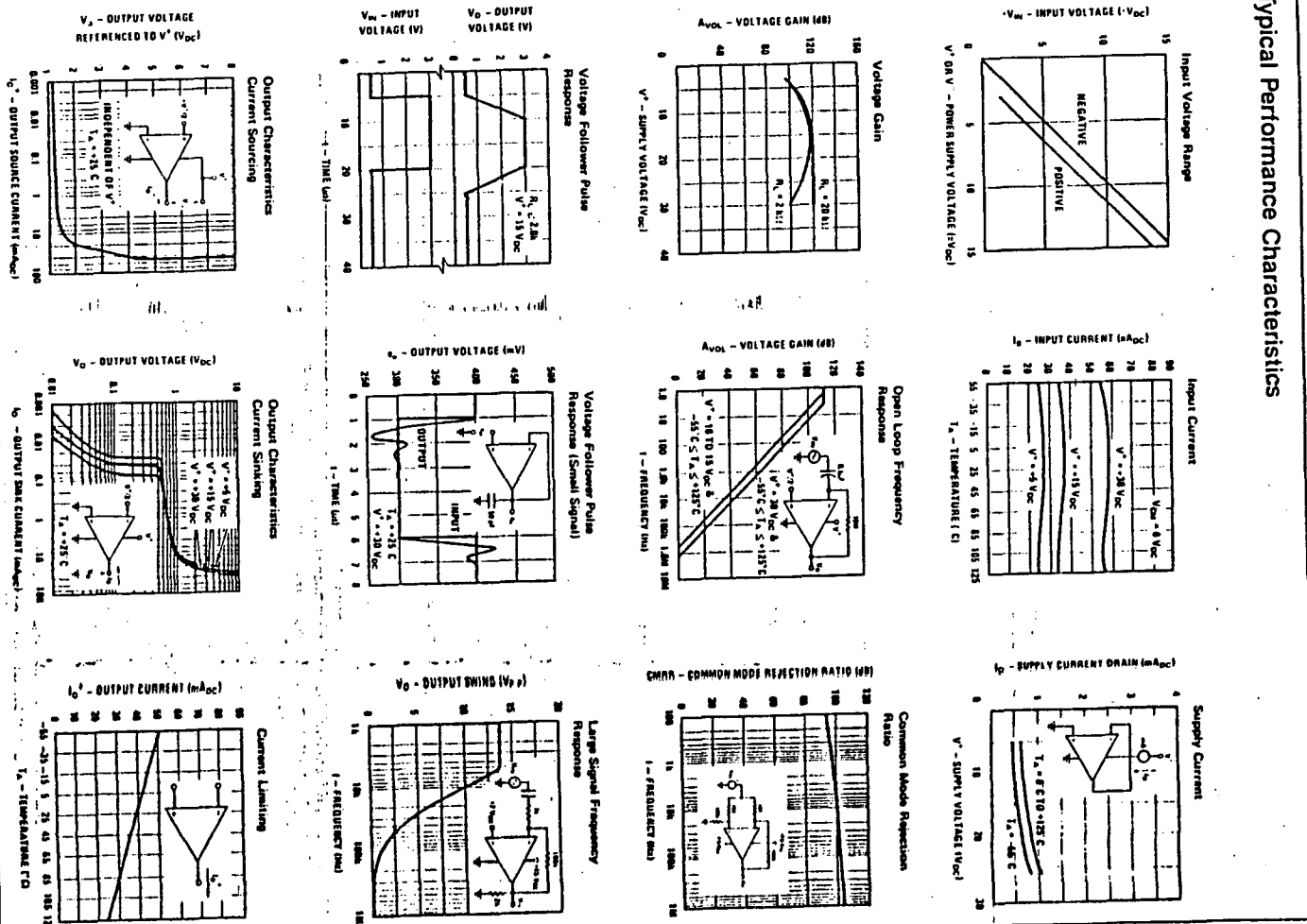
Note 5: $V_O \approx 1.4\text{VDC}$, $R_S = 0\Omega$ with V^+ from 5VDC to 30VDC ; and over the full input common-mode range (0VDC to $V^+ - 1.5\text{VDC}$).

Note 6: The direction of the input current is out of the IC due to the PNP input stage. This current is essentially constant, independent of the state of the output so no loading change exists on the input lines.

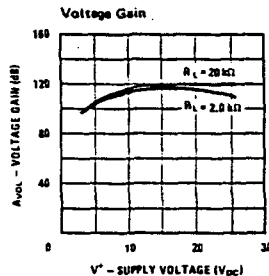
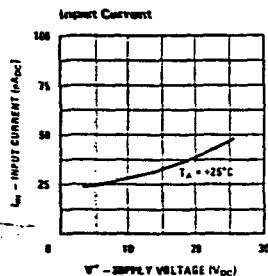
Note 7: The input common-mode voltage of either input signal voltage should not be allowed to go negative by more than 0.3V (at 25°C). The upper end of the common-mode voltage range is $V^+ - 1.5\text{V}$, but either or both inputs can go to $+32\text{VDC}$ without damage ($+26\text{VDC}$ for LM2902).

Note 8: Due to proximity of external components, insure that coupling is not originating via stray capacitance between these external parts. This typically can be detected as this type of capacitive increases at higher frequencies.

Typical Performance Characteristics



Typical Performance Characteristics (LM2902 only)



Application Hints

The LM124 series are op amps which operate with only a single power supply voltage, have true-differential inputs, and remain in the linear mode with an input common-mode voltage of 0 V_{DC} . These amplifiers operate over a wide range of power supply voltage with little change in performance characteristics. At 25°C amplifier operation is possible down to a minimum supply voltage of 2.3 V_{DC} .

The pinouts of the package have been designed to simplify PC board layouts. Inverting inputs are adjacent to outputs for all of the amplifiers and the outputs have also been placed at the corners of the package (pins 1, 7, 8, and 14).

Precautions should be taken to insure that the power supply for the integrated circuit never becomes reversed in polarity or that the unit is not inadvertently installed backwards in a test socket as an unlimited current surge through the resulting forward diode within the IC could cause fusing of the internal conductors and result in a destroyed unit.

Large differential input voltages can be easily accommodated and, as input differential voltage protection diodes are not needed, no large input currents result from large differential input voltages. The differential input voltage may be larger than V^+ without damaging the device. Protection should be provided to prevent the input voltages from going negative more than $-0.3 V_{DC}$ (at 25°C). An input clamp diode with a resistor to the IC input terminal can be used.

To reduce the power supply current drain, the amplifiers have a class A output stage for small signal levels which converts to class B in a large signal mode. This allows the amplifiers to both source and sink large output currents. Therefore both NPN and PNP external current boost transistors can be used to extend the power capability of the basic amplifiers. The output voltage needs to raise approximately 1 diode drop above ground to bias the on-chip vertical PNP transistor for output current sinking applications.

For ac applications, where the load is capacitively coupled to the output of the amplifier, a resistor should

be used, from the output of the amplifier to ground to increase the class A bias current and prevent crossover distortion. Where the load is directly coupled, as in dc applications, there is no crossover distortion.

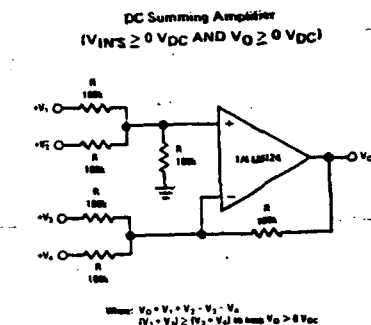
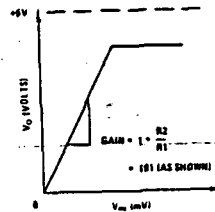
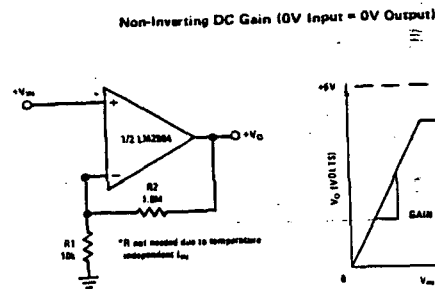
Capacitive loads which are applied directly to the output of the amplifier reduce the loop stability margin. Values of 50 pF can be accommodated using the worst-case non-inverting unity gain connection. Large closed loop gains or resistive isolation should be used if larger load capacitance must be driven by the amplifier.

The bias network of the LM124 establishes a drain current which is independent of the magnitude of the power supply voltage over the range of from 3 V_{DC} to 30 V_{DC} .

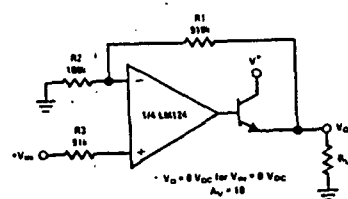
Output short circuits either to ground or to the positive power supply should be of short time duration. Units can be destroyed, not as a result of the short circuit current causing metal fusing, but rather due to the large increase in IC chip dissipation which will cause eventual failure due to excessive junction temperatures. Putting direct short-circuits on more than one amplifier at a time will increase the total IC power dissipation to destructive levels, if not properly protected with external dissipation limiting resistors in series with the output leads of the amplifiers. The larger value of output source current which is available at 25°C provides a larger output current capability at elevated temperatures (see typical performance characteristics) than a standard IC op amp.

The circuits presented in the section on typical applications emphasize operation on only a single power supply voltage. If complementary power supplies are available, all of the standard op amp circuits can be used. In general, introducing a pseudo-ground (a bias voltage reference of $V^+/2$) will allow operation above and below this value in single power supply systems. Many application circuits are shown which take advantage of the wide input common-mode voltage range which includes ground. In most cases, input biasing is not required and input voltages which range to ground can easily be accommodated.

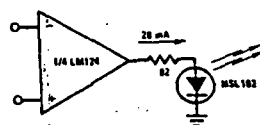
Typical Single-Supply Applications ($V^+ = 5.0 V_{DC}$)



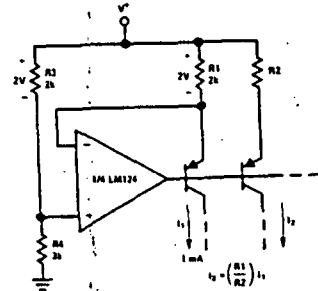
Power Amplifier



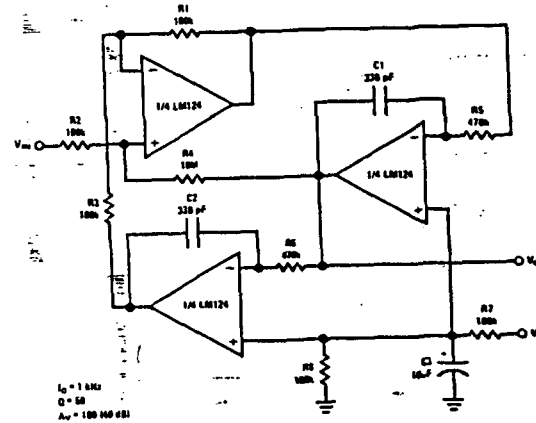
LED Driver



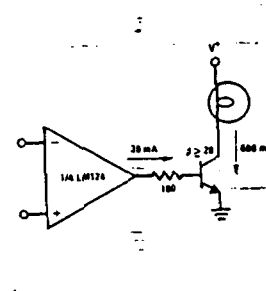
Fixed Current Sources



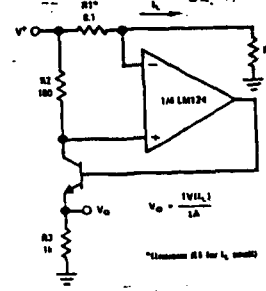
"BI-QUAD" RC Active Bandpass Filter



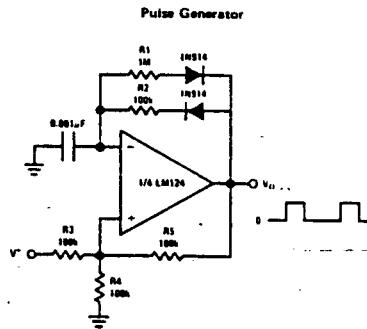
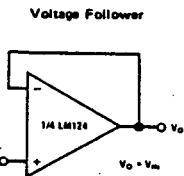
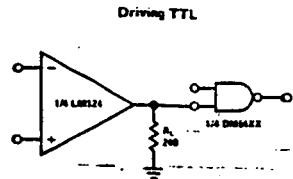
Lamp Driver



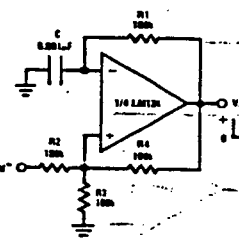
Current Monitor



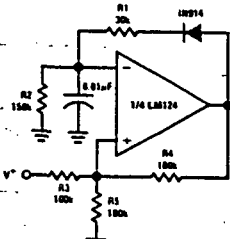
Typical Single-Supply Applications (Continued) ($V^+ = 5.0 V_{DC}$)



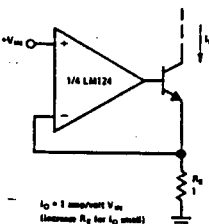
Squarewave Oscillator



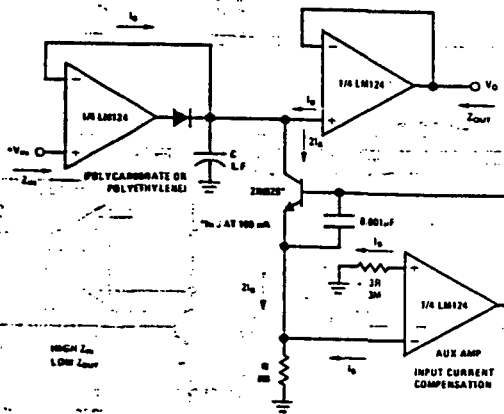
Pulse Generator



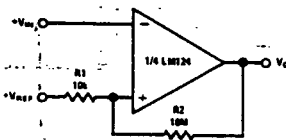
High Compliance Current Sink



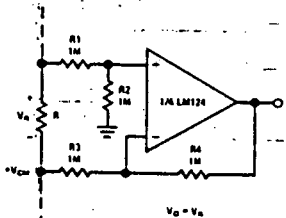
Low Drift Peak Detector



Comparator with Hysteresis



Ground Referencing A Differential Input Signal



Typical Single-Supply Applications (Continued) ($V^+ = 5.0 V_{DC}$)

Voltage Controlled Oscillator Circuit

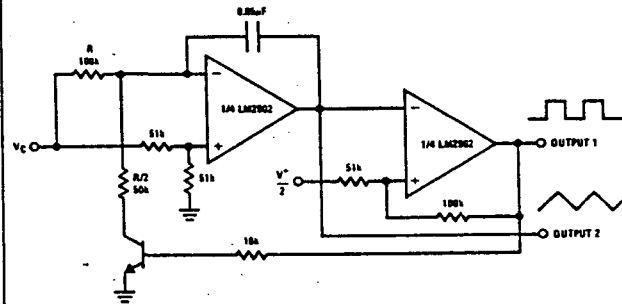
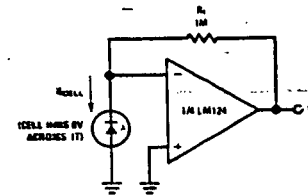
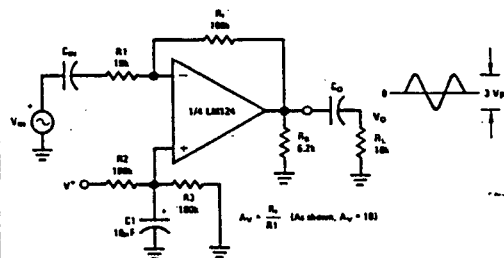


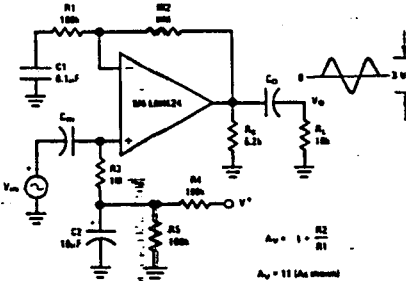
Photo Voltaic-Cell Amplifier



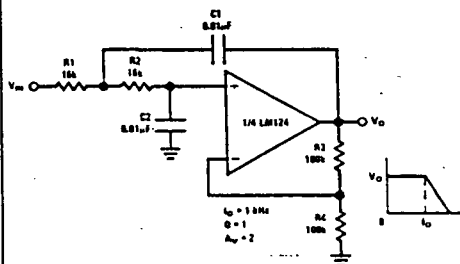
AC Coupled Inverting Amplifier



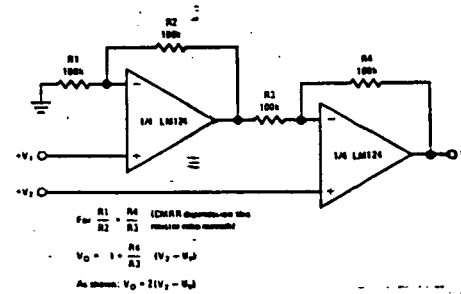
AC Coupled Non-Inverting Amplifier



DC Coupled Low-Pass RC Active Filter

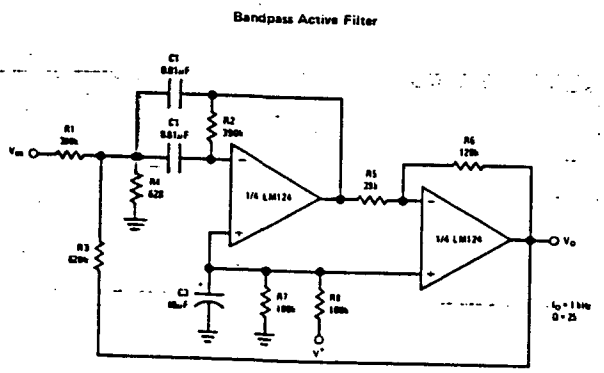
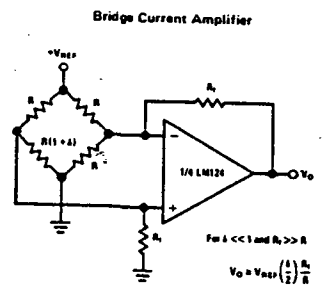
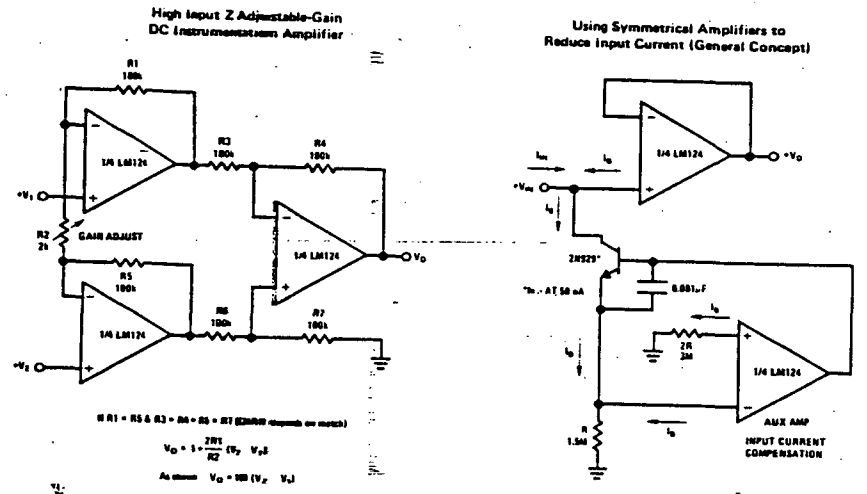


High Impedance DC-Differential Amplifier



LM124/LM324, LM2902

Typical Single-Supply Applications (Continued) ($V^+ = 5.0 V_{DC}$)



Operational Amplifiers/Buffers

LM143/LM343

LM143/LM343 High Voltage Operational Amplifier

General Description

The LM143 is a general purpose high voltage operational amplifier featuring operation to $\pm 40V$, complete input overvoltage protection up to $\pm 40V$ and input currents comparable to those of other super- β op amps. Increased slew rate, together with higher common-mode and supply rejection, insure improved performance at high supply voltages. Operating characteristics, in particular supply current, slew rate and gain, are virtually independent of supply voltage and temperature. Furthermore, gain is unaffected by output loading at high supply voltages due to thermal symmetry on the die. The LM143 is pin compatible with general purpose op amps and has offset null capability.

Application areas include those of general purpose op amps, but can be extended to higher voltages and higher output power when externally boosted. For example, when used in audio power applications, the LM143 provides a power bandwidth that covers the entire audio spectrum. In addition, the LM143 can be reliably operated in environments with large overvoltage spikes on the power supplies, where other internally-compensated op amps would suffer catastrophic failure.

The LM343 is similar to the LM143 for applications in less severe supply voltage and temperature environments.

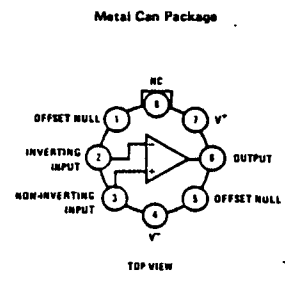
Features

- Wide supply voltage range $\pm 4.0V$ to $\pm 40V$
- Large output voltage swing $\pm 37V$
- Wide input common-mode range $\pm 38V$
- Input overvoltage protection Full $\pm 40V$
- Supply current is virtually independent of supply voltage and temperature

Unique Characteristics

- Low input bias current $8.0 nA$
- Low input offset current $1.0 nA$
- High slew rate—essentially independent of temperature and supply voltage $2.5V/\mu s$
- High voltage gain—virtually independent of resistive loading, temperature, and supply voltage $100k$ min
- Internally compensated for unity gain
- Output short circuit protection
- Pin compatible with general purpose op amps

Connection Diagram



Order Number LM143H or LM343H See NS Package H08C

B I B L I O G R A F I A

B I B L I O G R A F I A

- * "Microprocesadores y Lógica programada".

Kenneth L. Short.

Colección Electrónica-Informática. GS.

- * "Microprocesadores. Fundamentos, diseño y aplicaciones en la industria y en los minicomputadores".

J. M. Angulo.

Paraninfo.

- * "Materiales y Componentes Electrónicos".

R. Alvarez Santos.

- * "Revista Española de Electrónica".

- * "Manual del Emulador" del Sistema de Desarrollo MDS-221 de INTEL.

* "Circuitos Eléctricos III".

Elias Muñoz Merino.

E.T.S.I.T. Madrid.

* "Fiabilidad y mantenibilidad de sistemas".

Escuela de Informática de Madrid.