

UNIVERSIDAD DEL VALLE DE GUATEMALA

Facultad de Ciencias y Humanidades

Depto. de Ingeniería Electrónica

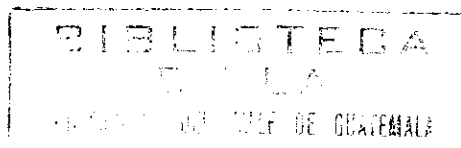


**DESARROLLO DE UNA INTERFAZ DE COMUNICACIÓN
ENTRE COMPUTADORAS PERSONALES Y UNA UNIDAD
DE ALIMENTACIÓN ININTERRUMPIDA**

MARIO ÁNGEL GARCIA CORTÉS

Trabajo de investigación presentado para optar
al grado académico de

Licenciatura en Ingeniería Electrónica



Guatemala

1997

A: Dios y la Virgen,
Por guiar e iluminar mis pasos

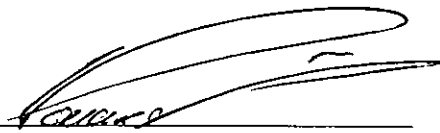
Mis padres, Mario y Petri

Mis hermanos, Rocío y Juan Manuel

Mi novia, Patty,
por todo su apoyo

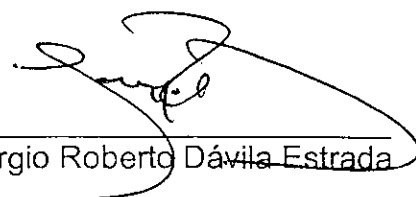
Mis jefes, Ing. Juan Carlos García e Ing. Héctor Urréjola,
por su apoyo y colaboración

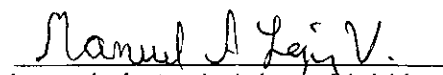
VO.BO.

(f) 
Ing. Gonzalo Palarea Murga
Asesor

TRIBUNAL EXAMINADOR

(f) 
Ing. Gonzalo Palarea Murga

(f) 
Ing. Sergio Roberto Dávila Estrada

(f) 
Dr. Manuel Antonio López Valdéz
Jefe de Departamento

Fecha de aprobación: 22 de octubre de 1997

INDICE

	Página
I. INTRODUCCIÓN	1
II. ANTECEDENTES	2
III. COMUNICACIÓN UPS-PC	17
IV. HARDWARE	24
V. SOFTWARE	31
VI. INSTALACION HARDWARE Y SOFTWARE	33
VII. CONCLUSIONES	35
VIII. BIBLIOGRAFIA	36
APENDICE	

LISTA DE TABLAS Y GRAFICAS

<u>Tabla</u>		<u>Página #</u>
1	Bus selector de Grupo de Led's	18
2	Combinaciones 7-segmentos-ALARMA	19
3	Conversión BCD-7 segmentos	20

I. INTRODUCCION

Los UPS (Uninterruptible Power Supply) son fuentes de poder de energía no interrumpidas, como su mismo nombre lo indica.

Los UPS marca Exide Electronics de la serie 1000 System, modelos S1100, S1200, S1300 y System 50, que se introdujeron en el mercado nacional a partir de 1985 son UPS's 100% en línea. Dichos UPS's, en los cuales la tarjeta de control basaba su funcionamiento en un microprocesador Z80, tienen la dificultad de no contar con una tarjeta de comunicación para poder ser controlado en forma automática desde una PC.

Al contar con una tarjeta de comunicación, el usuario podrá ir controlando de una forma sencilla y práctica el estado del UPS, así como ir almacenando un historial de eventos del UPS en un período determinado. También con dicha tarjeta de comunicación se puede revisar, en el instante que se desee, las mediciones respectivas tales como voltaje, corriente, frecuencia y alarmas activas.

II. ANTECEDENTES

A. UPS Series 1000

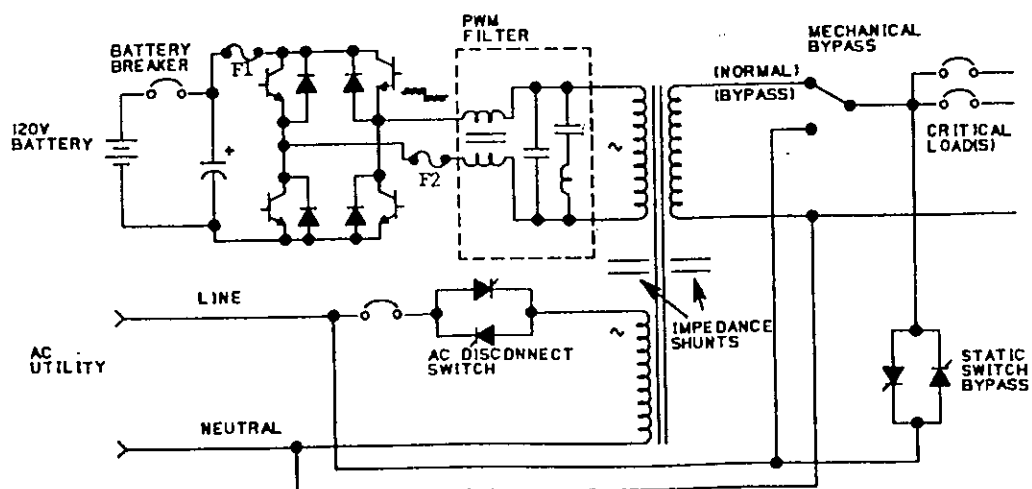
1. Descripción del Equipo

El UPS series 1000 es un cargador que usa un microprocesador bidireccional, y que controla un inversor transitorizado por medio de modulación por ancho de pulso. El transformador de enlace permite acoplar a la carga tanto “el bypass” como el inversor (fig.1). El inversor siempre se encuentra en fase con el voltaje de entrada o Empresa Eléctrica; la corriente es suministrada por la batería (cuando falta el voltaje de entrada) o la corriente es suministrada a la batería (rectificando la AC del voltaje de entrada). En el modo normal de operación del UPS, el voltaje AC de entrada proporciona la corriente a la carga y, cuando es necesario, recarga las baterías. Este diseño permite una regulación, tanto de voltaje como de forma de onda del voltaje de salida, en la que proporciona a la carga crítica (el equipo al cual proporciona energía el UPS) una forma de onda sin ningún tipo de distorsión armónica ni ruido, y con un valor RMS de 120 volts en todo momento, sin importar las variaciones que tenga el UPS en su alimentación. Cuando el voltaje de entrada falla, la línea de entrada es desconectada, y el inversor se alimenta de las baterías y así puede mantener la carga. Cuando regresa el voltaje de entrada el voltaje de salida del inversor se pone en fase con el voltaje de entrada o sea se sincroniza, y la corriente fluye del voltaje AC de entrada a la carga crítica y a las baterías.

Al existir una falla en el UPS, la línea de "bypass" es suministrada y conectada por medio de un contactor electromecánico, llamado interruptor estático, al voltaje de entrada y así se logra alimentar a la carga crítica. Si las baterías se encuentran muy descargadas al retornar el voltaje de entrada, éste es rectificado a través de los diodos que se encuentran en el paquete de transistores, efectuando así una recarga de emergencia. El inversor se mantendrá apagado y la carga crítica conectada a la línea de "bypass", hasta que las baterías estén suficientemente recargadas. Las baterías son selladas, libre de mantenimiento, con un tiempo de vida útil de 5 años, o 200 descargas totales. Las baterías son de 6 y 12 volts, con una capacidad de 10, 24 ó 38 amperios/hora.

B. Teoría de Operación

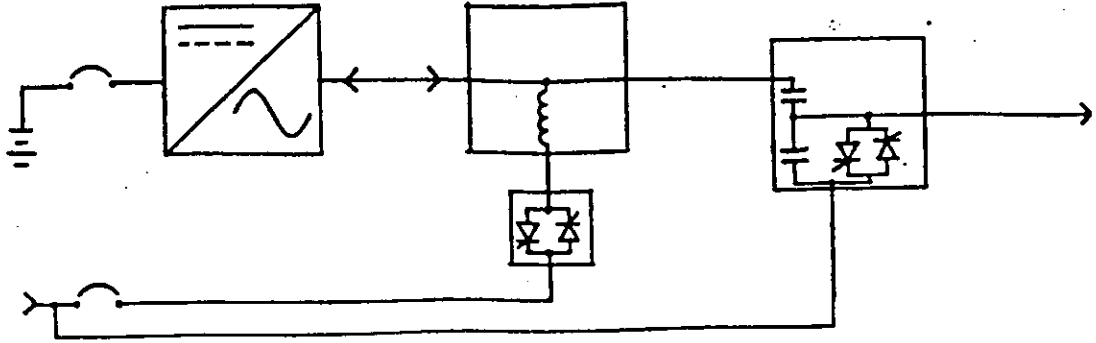
Un diagrama simplificado de un UPS Interactivo es mostrado en la Fig 1. Es utilizado un transformador lineal con tres devanados de aislamiento. El inversor es acoplado a la carga crítica por medio de los devanados , controlando el voltaje de la carga, forma de



onda y supliendo la demanda de corriente para cargas no lineales cuidando el límite o capacidad del inversor. El voltaje AC de entrada es acoplado débilmente al voltaje AC de la carga por medio de derivaciones magnéticas en el transformador (esto permite tener efectivamente el inversor y la carga en paralelo, pero atrasada en fase con la entrada a través de un inductor en serie). Cuando el inversor opera de manera sincronizada con el voltaje de entrada, el flujo de poder está en función de la variación del ángulo de fase entre el voltaje del inversor y el voltaje de la entrada. Cuando el ángulo de fase es pequeño, el inversor proporcionará parte de los requerimientos de la carga, y las baterías serán descargadas. Como el ángulo de fase se va incrementando, el inversor suplirá menos potencia a la carga (y la alimentación del UPS suplirá más). En modo normal de operación el inversor regula el voltaje de la carga, pero la alimentación de entrada proporciona la potencia a la carga crítica. La onda senoidal AC es generada al usar un puente inversor transistorizado. Los pulsos de PWM (Pulse Width Modulator) de salida están acoplados a través de un filtro pasabajos al devanado del inversor del transformador. El control de la forma de onda es conseguido por comparación del voltaje AC de salida contra un generador de onda senoidal usada como referencia. El voltaje de entrada es conectado al devanado del transformador por medio de un interruptor de AC, el cual consta de 2 SCR. El modo normal de operación, el inversor controla la forma de onda de salida, y el voltaje de alimentación proporciona la potencia necesaria para la carga. Si el voltaje de alimentación falla, el interruptor AC se abre, y el inversor proporciona toda la potencia necesaria para la carga, la cual se obtiene de sus baterías.

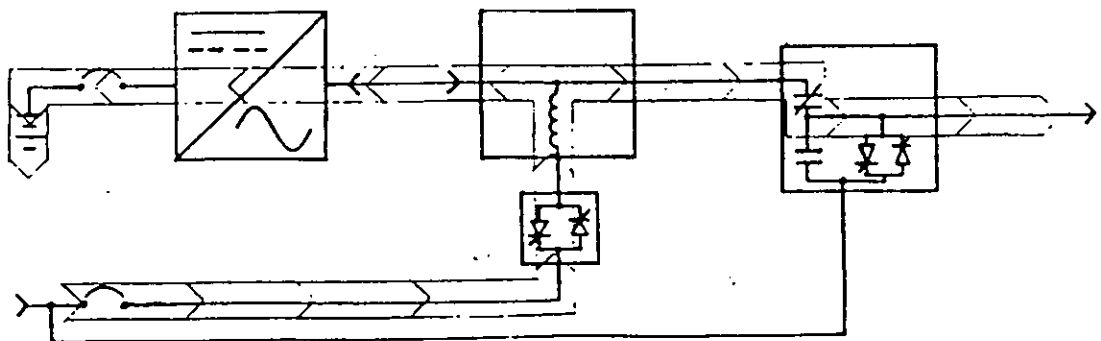
C. Diagrama de Bloque

La Fig. 2 es un diagrama simplificado de un UPS Series 1000.

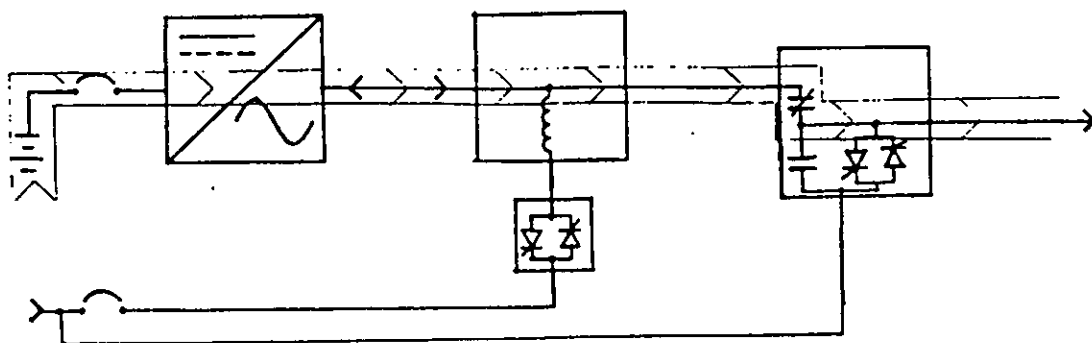


1. Modos de operación

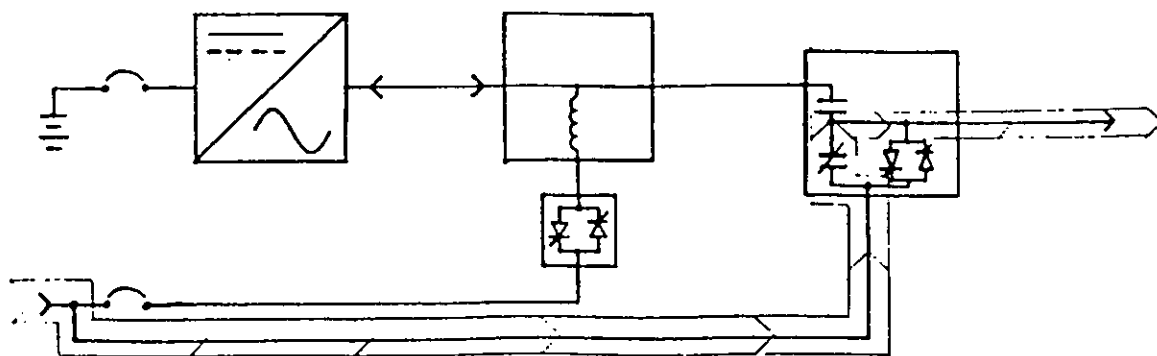
a) Operación Normal: La Figura 3 muestra cómo el flujo de potencia es proporcionado por el voltaje de alimentación hacia el inversor, baterías y la carga crítica.



- b) **Falla Voltaje de Entrada:** La Fig. 4 muestra cuando el voltaje de entrada falla y, el inversor proporciona todos los requerimientos de la carga.

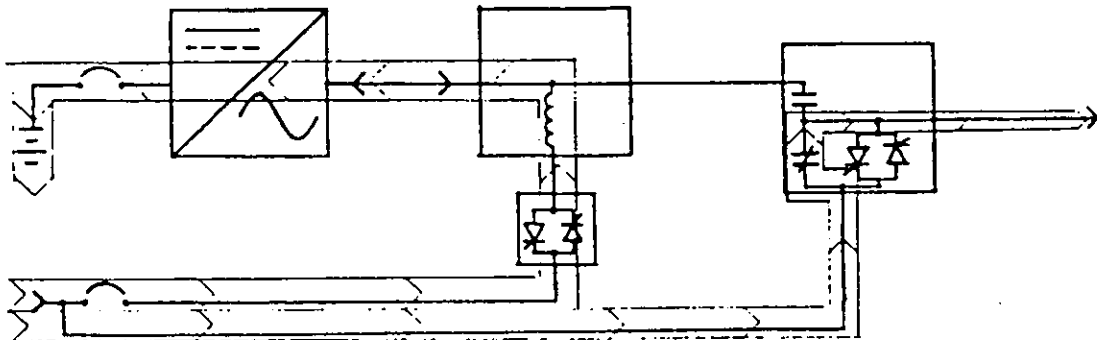


- c) **Falla Inversor:** La Figura 5 muestra cuando el inversor falla y, la carga es proporcionada a través de línea de bypass. Si el interruptor estático es usado, éste mantiene la carga hasta que el contactor se cierre.



d) Recarga de Emergencia: La figura 6 muestra cuando las baterías se encuentran totalmente descargadas, y el voltaje de entrada es conectado, entonces el inversor intenta encender. Si este último no puede encender, el switch de AC se conecta, y la corriente fluye a través de los diodos del paquete de transistores para recargar las baterías. Durante este tiempo, la carga es alimentada por la línea de bypass.

Después de 5 minutos de recargarlas, el inversor intentará arrancar y entrar en modo normal.



D. Descripción de partes

Las Series 1000 tienen incorporadas 3 tarjetas, las que son el alma del UPS. Estas tres tarjetas son la tarjeta de control, la fuente de poder y la tarjeta de potencia, interconectadas por medio de una tarjeta madre.

1. Tarjeta de Control

Consiste de un CPU de 8 bits (Z280), tres chips decodificadores de direcciones, tres contadores programables de tiempo (CTC), 1K byte de RAM, 4K byte de ROM, un DAC de 8 bits, un ADC de 8 bits y líneas de entrada/salida usadas para control y despliegue de datos (Fig. 7)

2. Tarjeta de Potencia

Se compone de cuatro canales independientes de transistores de potencia del inversor. La señal de PWM proveniente de la tarjeta de control es aislada ópticamente por la tarjeta de potencia, para evitar ruido entre potencia y control (Fig. 8).

3. Fuente de Poder

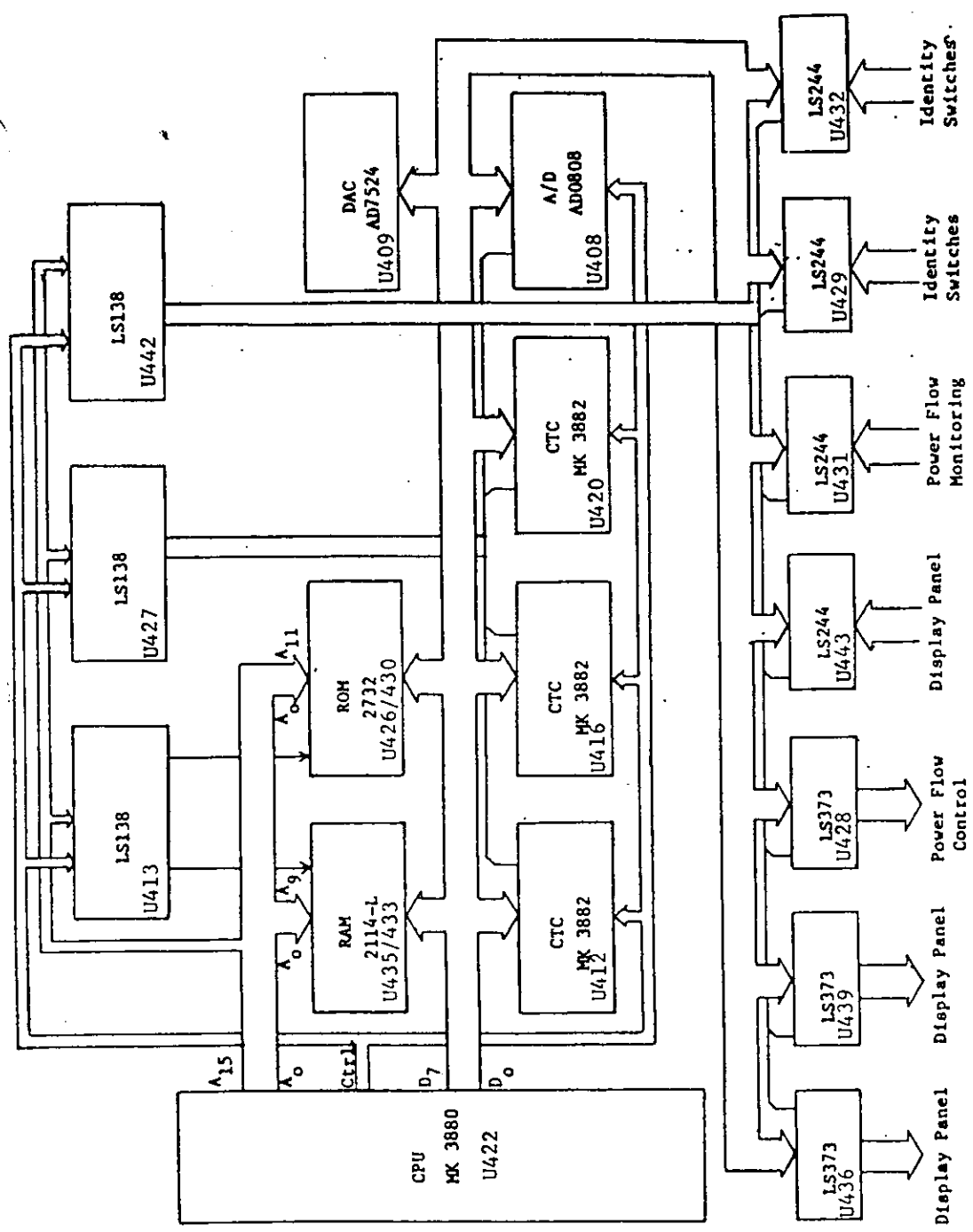
Da alimentación a toda la parte de control de UPS y la tarjeta de potencia. La fuente de poder tiene voltajes de salida de -12 volts, 15 volts, 5 volts, además tiene 3 fuentes variables de 8.5-12 volts, que alimentan los 4 canales de la tarjeta de potencia (Fig. 9)

4. Panel del Despliegue

Consiste de un manejador universal de despliegue por medio de grupos de 8 leds, un codificador BCD de 10 a 4, 3 dígitos de 7 segmentos, 15 LED tipo barra, una alarma audible y 9 interruptores tipo botón.

Dos Flip-Flop tipo-D de 8 bits en la tarjeta de control controlan el manejador de LED U501 (referirse a la Fig. 10). Las líneas A0-A5 provenientes del CPU controlan y seleccionan al U501, U504 es una compuerta NOR, la cual es usada para indicar qué tipo de datos se están recibiendo.

Al presionar alguno de los 9 botones o switches, los cuales están conectados directamente al codificador de 10 a 4 U502, la tarjeta de control reaccionará y enviará la información requerida y posteriormente la tarjeta de control lo trasmitirá a U501 de la tarjeta de despliegue (Fig. 10).



CONTROL BOARD PROCESSOR BLOCK DIAGRAM

Figura 7

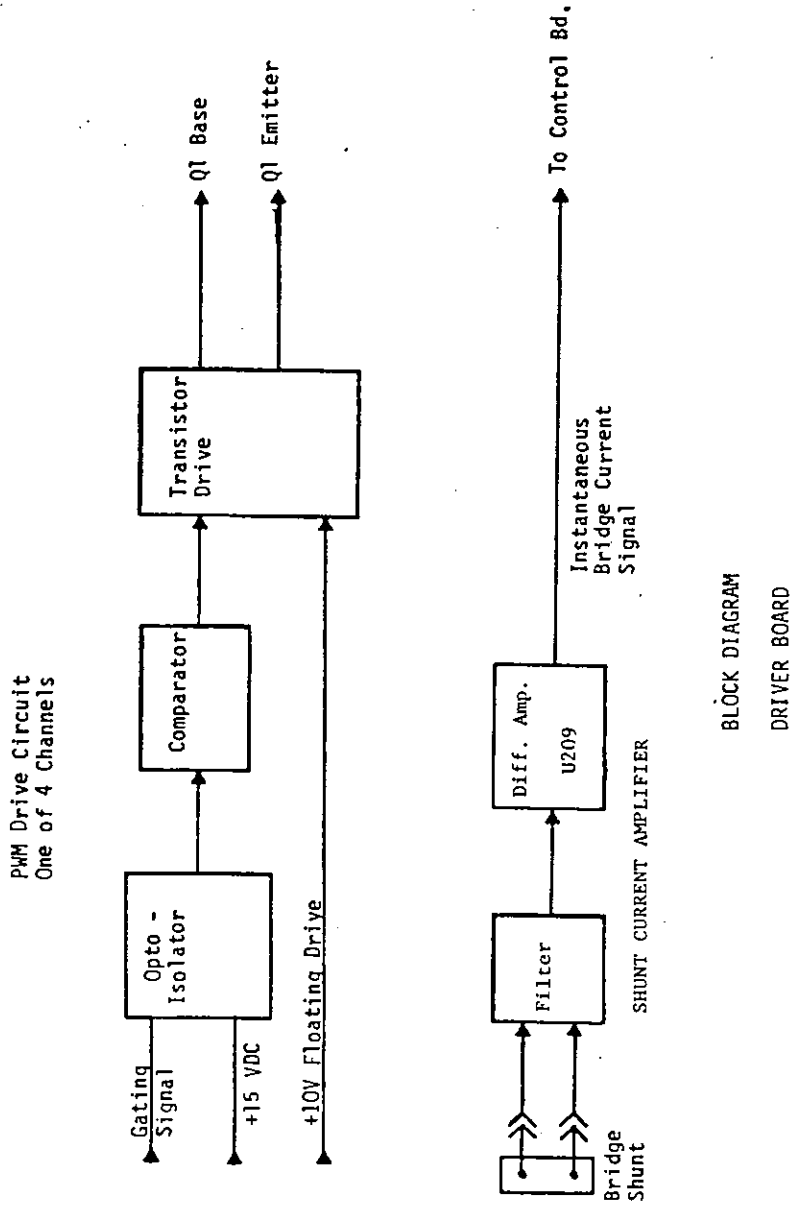
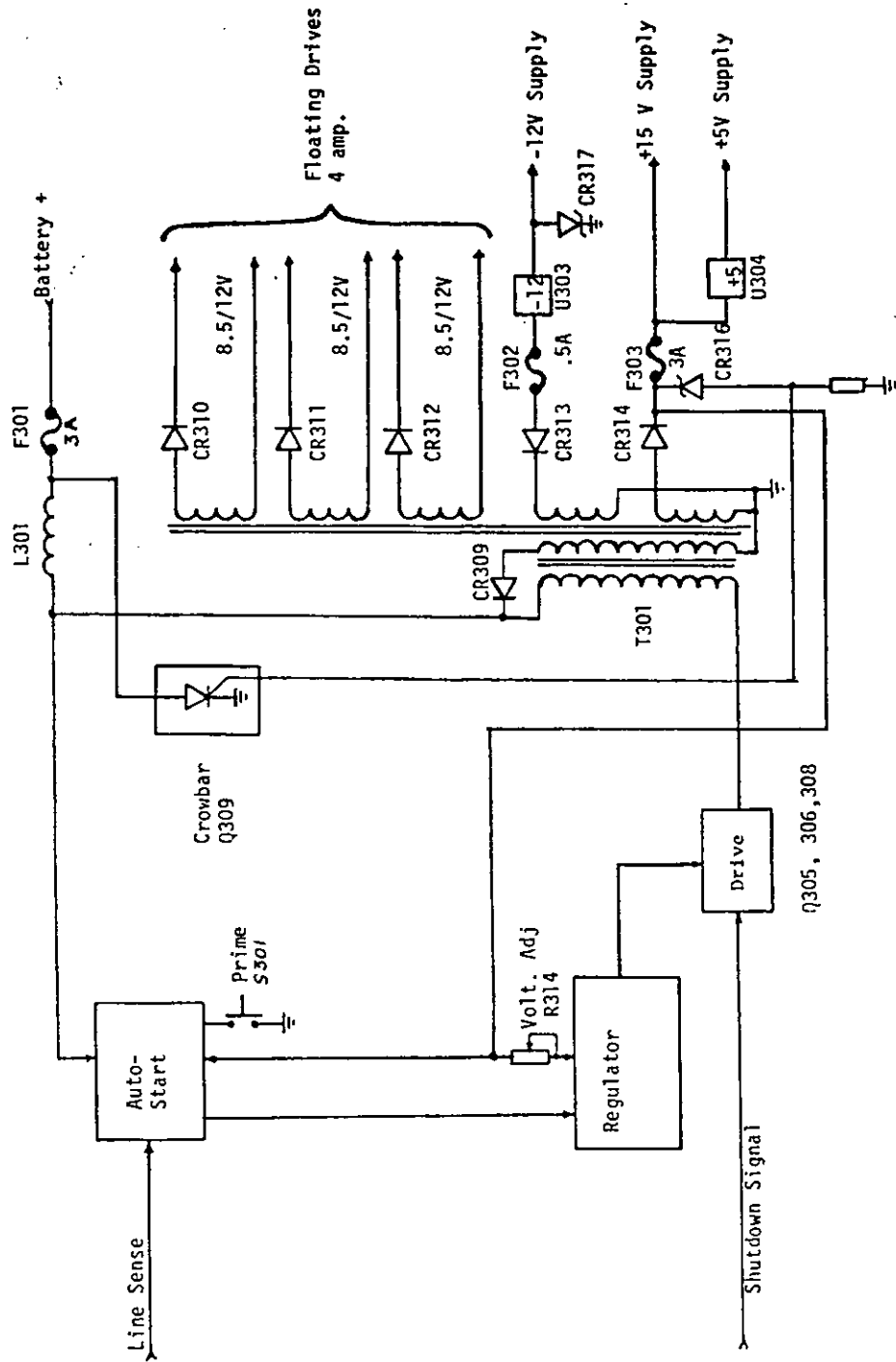
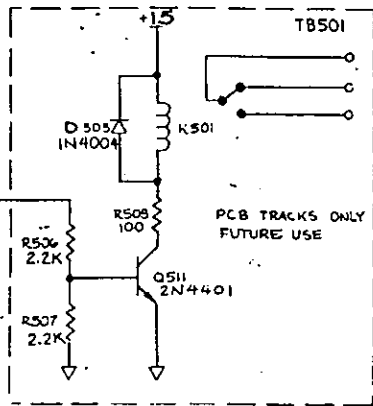


Figura 8



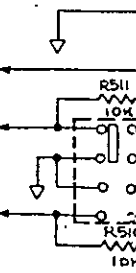
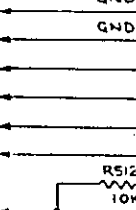
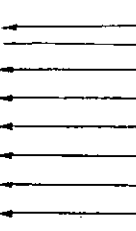
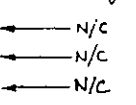
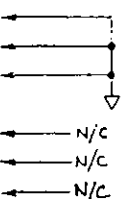
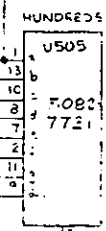
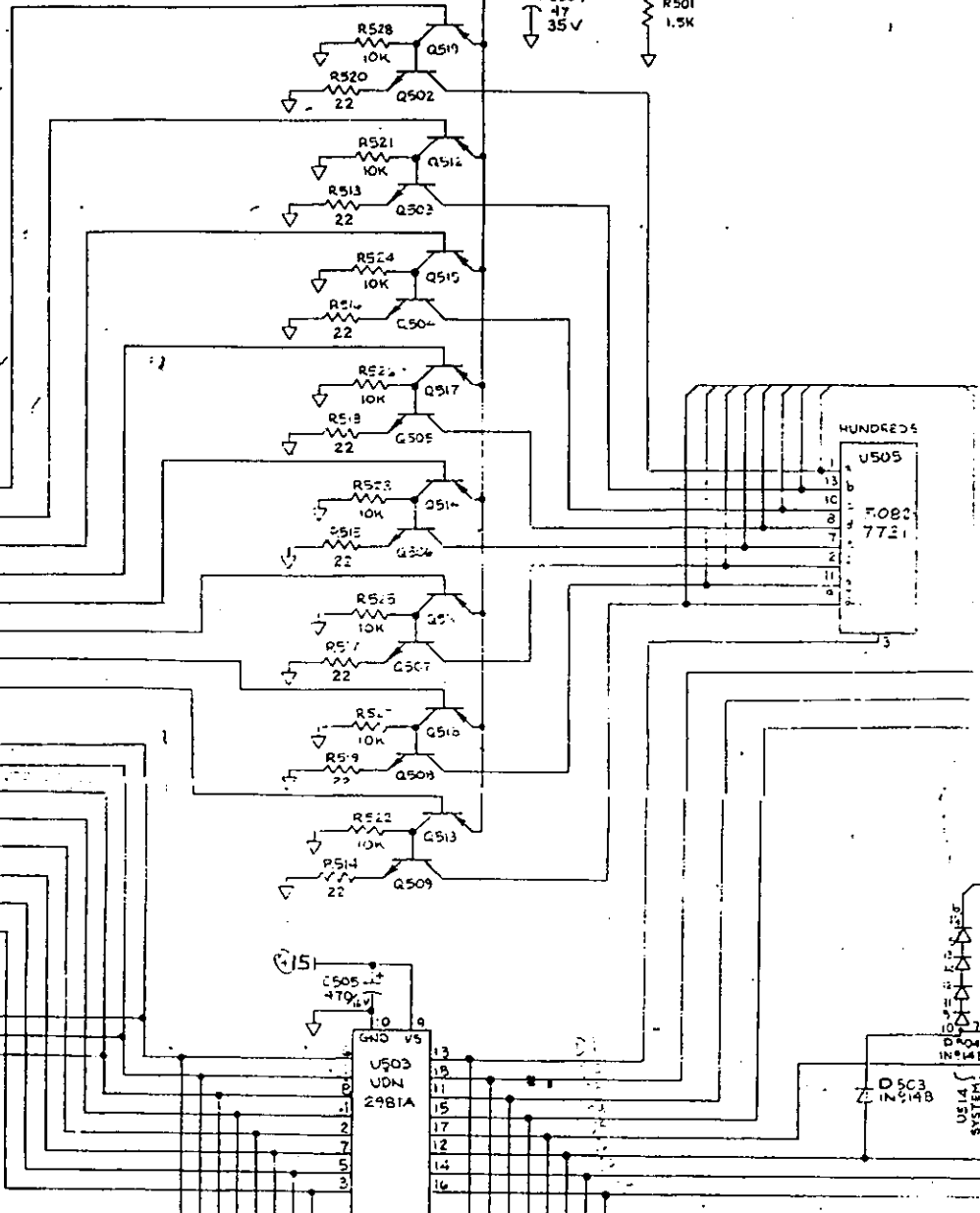
BLOCK DIAGRAM
POWER SUPPLY BOARD

Figura 9



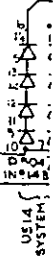
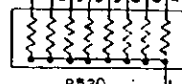
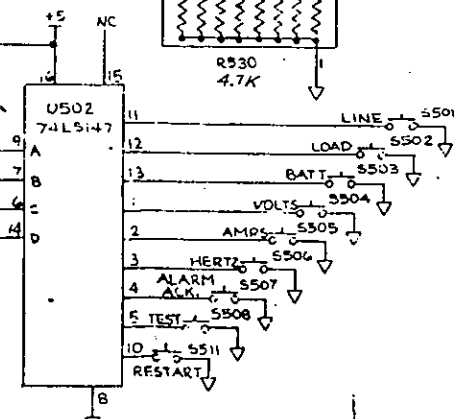
DISPLAY BRIGHTNESS

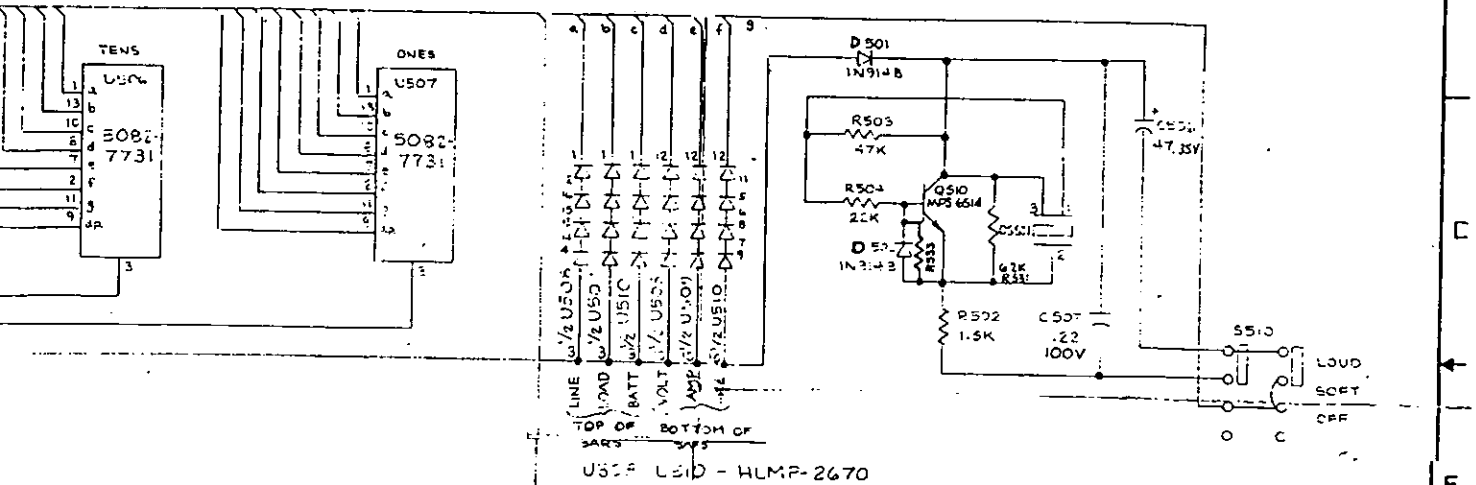
Q502-509 2N4401
Q512-519 2N403



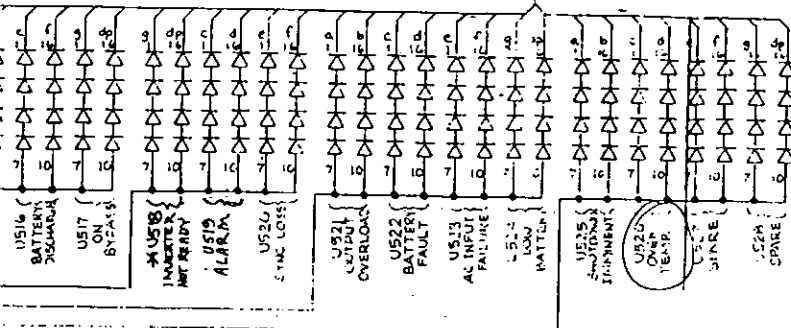
LAST USE

- R533
- C507
- Q519
- D 505
- U528
- D5501
- K501
- S512
- J503
- L501
- TB501





TOP OF BOTTOM OF
 U504 LED - HLMP-2670



U514 - HLMP-2853
 U515-U526 - HLMP-2-58

NOTES:

1. ALL RESISTORS ARE 1/4 WATT CARBON, EXCEPT AS NOTED.
 2. ALL CAPACITANCES ARE IN MICROFARADS.
- * WITH J503-1 AND J503-2 CONNECTED TOGETHER THE ALARMS ARE AS SHOWN; WITH U503-1 AND J503-2 OPEN, U518 IS CHARGER FAILURE, U519 IS LOAD OFF, AND U515 IS INVERTER FAILURE

NOT USED
 R506, 507, 508
 Q511
 D505
 U511, 512, 513
 K501
 S509, 512
 J502
 TB501

INFORMATION HEREIN IS PROPRIETARY & CONFIDENTIAL EXIDE ELECTRONICS CORPORATION

EXIDE ELECTRONICS

Unless otherwise specified - Fabrication Decimals Fractions Angles		Date Part No. Rev. Date	Computer Description 25 Characters Max. (1-10) 110 071 458 DISPLAY PANEL SCHEMATIC S1100 (101 072 612)
B CN63-093 A CN-63-031	Dimensional are in inches and other plating Do not scale drawing This drawing and specifications are the property of Exide Electronics and shall not be reproduced or copied, or used in the manufacture of parts for sale or other work without written permission.	Part No. 110 071 458 Custom No.	Sheet 1

A. LISTADO DE ALARMAS

A continuación se presenta un listado de todas las alarmas que pueden presentar UPS series System, con una breve descripción de lo que significa cada una:

- **System Normal:** La unidad ha ordenado cerrar el contactor, y el inversor se pone en línea con la carga crítica; las baterías y el voltaje de salida se encuentran con los valores deseados.
- **Alarm:** Esta suena cuando alguna alarma está presente.
- **Low Battery:** El voltaje de baterías es menor a 128 volts DC. Dicha alarma desaparece hasta que el voltaje de las baterías sea mayor a 132 volts.
- **Battery Discharges:** Cuando las baterías descargan una corriente mayor 2.5 amperios DC, dicha alarma se activa.
- **Output Overload:** La corriente que demanda la carga crítica es mayor al 103% de su capacidad. Si la sobrecarga persiste por más de 10 minutos, el UPS se transferirá a Bypass.
- **Sync Loss:** El voltaje de entrada o alimentación ha sobrepasado la ventana de frecuencia de 3 Hz. El UPS acepta una tolerancia de frecuencia de entrada de 57-63 Hz.
- **AC Input Failure:** El voltaje de alimentación del UPS ha sobrepasado los 65° C. El UPS se transferirá a Bypass.
- **Shutdown Inminent:** El inversor se apagará si las condiciones existentes persisten. Esta alarma puede aparecer en el momento en que el voltaje de las baterías sea menor a 110 volts o después de tener una sobrecarga mayor al 106%, durante 8 minutos. La carga se transferirá a Bypass cuando el voltaje de las baterías llegue a ser 102 volts, o la sobrecarga (106%) dure más de 10 minutos.
- **On Bypass:** La carga recibe toda la potencia de la línea de bypass (voltaje AC de entrada).

- Inverter Not Ready: La carga se encuentra conectada a la línea de bypass y el voltaje de salida del inversor aún no se ha estabilizado. Esta alarma indica una distorsión de 5% en la forma de onda de salida del inversor.
- Inverter Failure: Dicha alarma nos indica que la forma de onda de salida del inversor está total o parcialmente distorsionada, debido a que los transistores de potencia se encuentran dañados o a que algún canal de la tarjeta de potencia se encuentra quemado. También podría provocar que algún condensador del filtro pasabajos no esté trabajando bien.
- Battery Fault: El voltaje de las baterías es menor a 102 Volts DC.

F. DIAGRAMA FASORIAL DEL FLUJO DE POTENCIA.

La Fig. 11 es un circuito equivalente de un UPS Series 1000. Este consiste de un inversor acoplado al voltaje de entrada (Utility) a través de un inductor L_s con la carga conectada directamente a través del inversor. El voltaje del inversor E_i se encuentra desfasado por un ángulo B . La corriente del inductor I_L se adelanta en 90 grados al voltaje del inductor E_L . La corriente del inversor I_L representa el vector de diferencia entre I_o y I_L .

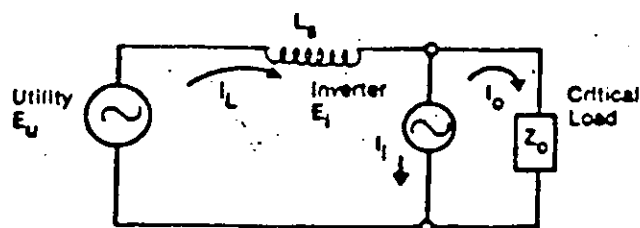
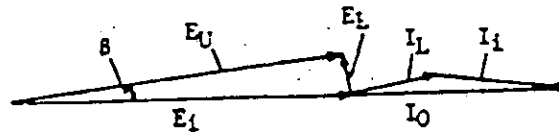


Figura 11

Cuando B es pequeño, la corriente del inductor I_L es también pequeña. La corriente del inversor se encuentra casi en fase con el voltaje del Inversor E_i y por consiguiente el Inversor está entregando toda la potencia real requerida por la carga (la entrada no entrega potencia). En este caso, las baterías están siendo descargadas (Fig. 12).



$$E_u = E_i$$

$$pf = 1.0$$

Figura 12

Cuando B se incrementa la parte real de I_L se iguala a I_0 , y la Entrada (Utility) proporciona toda la potencia a la carga. Como I_i está fuera de fase con E_i , no fluye potencia real hacia adentro o afuera del Inversor, en este momento la corriente DC de baterías es cero (Fig 13)



Figura 13

Para un gran desplazamiento del ángulo de fase B, la dirección del vector I_1 indica que la potencia real fluye hacia adentro del Inversor, y las baterías serán cargadas.

El voltaje de entra E_u (Utility) está supliendo los requerimientos de la carga. (Figura 14).

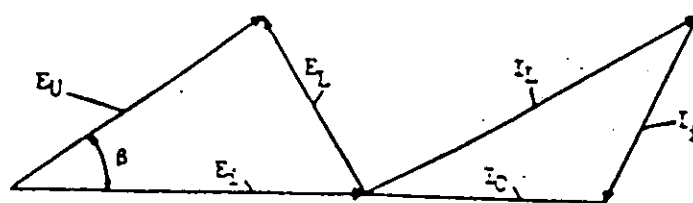


Figura 14

III. COMUNICACIÓN UPS-PC

Para poder obtener cualquier tipo de información del UPS, se debió de realizar una unidad que conectara las señales existentes en el panel de despliegue con la Computadora.

El panel de despliegue contiene un manejador universal para LED's (U501), el cual selecciona el estado de los 8 bits de entrada, transfiriéndolos al bus de 7 segmentos o al bus de selección de LED's.

Los dos buses de salidas de 8 bits del U501 manejan todos los LED's del panel de despliegue, tanto los despliegues de datos numéricos como los despliegues de las alarmas respectivas. El bus de las salidas de 7 segmentos (a,b,c,d,e,f,g,dp) se encuentra conectado a todos los LED's del Panel. El segundo bus de salida del U501 es el de selección de LED's (D1,D2,D3,D4,D5,D6,D7) es utilizado para seleccionar qué grupo de LED's se encienden.

Esto último se puede visualizar mejor en la Tabla 1 donde se muestran todos los posibles valores del bus de 7 segmentos y del bus de selección de LED's.

En la Tabla 2 se observa todos los posibles valores del bus de 7 segmentos, para los cuales aparecerían las alarmas respectivas. Por ejemplo si el bus de datos de selección tiene un valor DEC de 16, y el bus de datos de 7 segmentos tiene un valor DEC de 248 el LED que llegaría a encenderse sería el de NORMAL; si el bus de datos de selección tiene un valor DEC de 16 pero el bus de datos de 7 segmentos tiene un valor DEC de 63 el LED que se encendería sería el INVERTER NOT READY; y así sucesivamente.

Todos los botones del panel de despliegue se encuentran conectados a la entrada del codificador de 10 a 4 U502 del panel; si no se oprime ningún botón, el decodificador se tiene 0 volts en todas sus entradas. Los 4 bits de salida hacen un requerimiento a la

TABLA I

Dependiendo del valor de D1-D8, se selecciona un grupo de leds a encender:

D1	D2	D3	D4	D5	D6	D7	D8	GRUPO LED'S
1	0	0	0	0	0	0	0	Centenas-7 segmentos
0	1	0	0	0	0	0	0	Decenas-7 segmentos
0	0	1	0	0	0	0	0	Unidades-7 segmentos
0	0	0	1	0	0	0	0	Output overload, Battery Fault, AC Input Failure, Low Battery
0	0	0	0	1	0	0	0	Inverter Not Ready, Sync Loss
0	0	0	0	0	1	0	0	Inverter Failure, On Bypass Battery Discharge
0	0	0	0	0	0	1	0	
0	0	0	0	0	0	0	1	Overtemp, Shutdown Inminent

TABLA 2

Combinaciones de bits presentes en el bus de 7 segmentos.

A	B	C	D	E	F	G	.dp	ALARMA
0	0	1	1	1	1	1	1	NORMAL
1	1	0	0	1	1	1	1	INVERTER FAILURE
1	1	1	1	0	0	1	1	BATTERY DISCHARGE
1	1	1	1	1	1	0	0	ON BYPASS
1	1	1	1	1	1	0	0	INVERTER NOT READY
1	1	1	1	0	0	1	1	SYNC LOSS
0	0	1	1	1	1	1	1	OUTPUT OVERLOAD
1	1	0	0	1	1	1	1	BATTERY FAULT
1	1	1	1	0	0	1	1	AC INPUT FAILURE
1	1	1	1	1	1	0	0	LOW BATTERY
0	0	1	1	1	1	1	1	SHUTDOWN IMMINENT
1	1	0	0	1	1	1	1	OVERTEMP

TABLA 3

En la siguiente tabla se tiene la conversión BCD-7-segmentos.

A	B	C	D	E	F	G	.dp	No.
1	0	0	1	1	1	1	1	1
0	0	1	0	0	1	0	1	2
0	0	0	0	1	1	0	1	3
1	0	0	1	1	0	0	1	4
0	1	0	0	1	0	0	1	5
0	1	0	0	0	0	0	0	6
0	0	0	1	1	1	1	1	7
0	0	0	0	0	0	0	1	8
0	0	0	1	1	0	0	1	9
0	0	0	0	0	0	1	1	0

Los 4 bits de salida hacen un requerimiento a la tarjeta de control de la información que se necesita. La tarjeta de control manda la información al panel de despliegue, y U501 lo clasifica.

Sabiendo cómo es el manejo de datos entre el Z80 de la tarjeta de control y el panel de despliegue, que U501 es el corazón de toda esta operación, que es necesario tener un control del teclado del UPS para poder pedir información numérica, se necesita un interface que maneje los buses de datos y el teclado.

Por lo anterior se escogió un periódico que fuera capaz de manejar tres puertos para todos los datos ya mencionados. Para esto se escogió el INTEL 8255.

INTEL 8255

Es un interface periférico programable de propósito general, que proporciona 24 pines de entrada/salida, los cuales pueden ser configurados como uno, dos o tres puertos.

El 8255 es empacado en una cápsula de 40 pines. Usa una única alimentación de +5V. Todas las entradas y salidas son compatibles con niveles de TTL. El dispositivo es implementado al usar tecnología Canal-n MOS.

Pines y Señales

DO – D7: representa el bus de datos bidireccional, cualquier comunicación entre el CPU y el 8255 ocurre por esta vía.

PA0 – PA7, PB0 – PB7, PC0 – PC7: representa el bus de datos que está conectado a cualquiera de los tres puertos de entrada A, B, C de 8 bits. Toda comunicación con lógica externa se realiza en forma paralela por alguno de estos tres puertos.

CS es el chips select cuando hay un cero lógico en CS, el 8255 es seleccionado, A0 y A1 permite direccionar a cualquiera de los 4 registros que tiene internamente el 8255. La combinación de A0 y A1 permite lo siguiente:

CS	A1	A0	SELECCIÓN
0	0	0	Puerto A
0	0	1	Puerto B
0	1	0	Puerto C
0	1	1	Palabra de Control

RD y WR se activan con un cero, que selecciona operaciones de lectura y escritura con los registros del 8255. Cuando RD es cero, los contenidos del registro seleccionado son leídos por el CPU. Cuando WR es cero, el CPU escribe en el registro seleccionado.

Modos de Operación del 8255

El 8255 tiene cuatro localizaciones direccionales, estas son tres puertos tanto de entrada como de salida y un registro para la palabra de control. El Puerto C puede ser dividido en dos puertos de 4 bits, ingresados por una simple dirección. Los puertos son ingresados como tres unidades paralelas de 8-bit independientes.

El 8255 tiene tres modos de operación distintos que son el modo 0, el modo 1, el modo 2.

Modo 0: En este modo cada puerto entrada/salida puede ser definido como un simple puerto de entrada o un simple puerto de salida.

Modo 1: En este modo se permite al Puerto A y B ser definido como puerto de entrada o de salida, con el Puerto C al proporcionar señales de "handshaking". El puerto A puede ser asignado como entrada o salida, pero no ambos. La parte alta del puerto C sirve para señales de control del Puerto A. El Puerto B puede ser asignado como entrada o salida, pero no en ambas. La parte baja del puerto C sirve para señales de control del puerto B.

Modo 2: Este modo permite actuar el puerto A como un puerto de datos bidireccional. Las señales de control de handshaking bidireccionales son proporcionadas por el puerto C.

La palabra de control del 8255 es interpretada de la siguiente forma:

No. Bit

0 :Puerto C parte baja 0 = salida, 1 = entrada

1 :Puerto B 0 = salida, 1 = entrada

2 :Puerto B 0 = Modo 0, 1 = Modo 1

3 :Puerto C parte alta 0 = salida, 1 = entrada

4 :Puerto A 0 = salida, 1 = entrada

5,6 :Puerto A00 = Modo 0

01 = Modo 1

10 = Modo 2

11 = Modo 2

7 :1

IV. HARDWARE

Para la realización del interface se elaboraron dos tarjetas para la comunicación paralela entre el UPS y la PC. La primer tarjeta es una manejadora de puertos, en la cual se tiene al 8255 como el periférico responsable para dicha función. Esta tarjeta va colocada internamente en la PC.

La segunda tarjeta, que se instaló en el UPS, se encarga de cumplir dos funciones:

- Aislamiento óptico entre el UPS y la PC, para no tener interferencia entre las tierras respectivas,
- Utilización de buffer para no cargar el bus de datos del UPS como el bus de datos de la PC.

TARJETA MANEJADORA DE PUERTOS

Contiene un 8255 que conecta internamente en la PC en cualquier slot que esté disponible. La tarjeta se encuentra dirigida como puerto de entrada/salida, en la dirección 0300h.

El 8255 se configuró para trabajarlo en el modo 0, de tal forma que el puerto A y el puerto C se usan como puertos de entrada y el puerto B como puerto de salida.

El puerto A se utilizó para capturar al bus de datos de selección de grupo de LED's del U501, el puerto C se utilizó para capturar el bus de datos de 7-segmentos de U501. El puerto B se utiliza como puerto de salida, para usar el teclado de la tarjeta de potencia del UPS y mandar los códigos binarios necesarios para eso (Fig. 15).

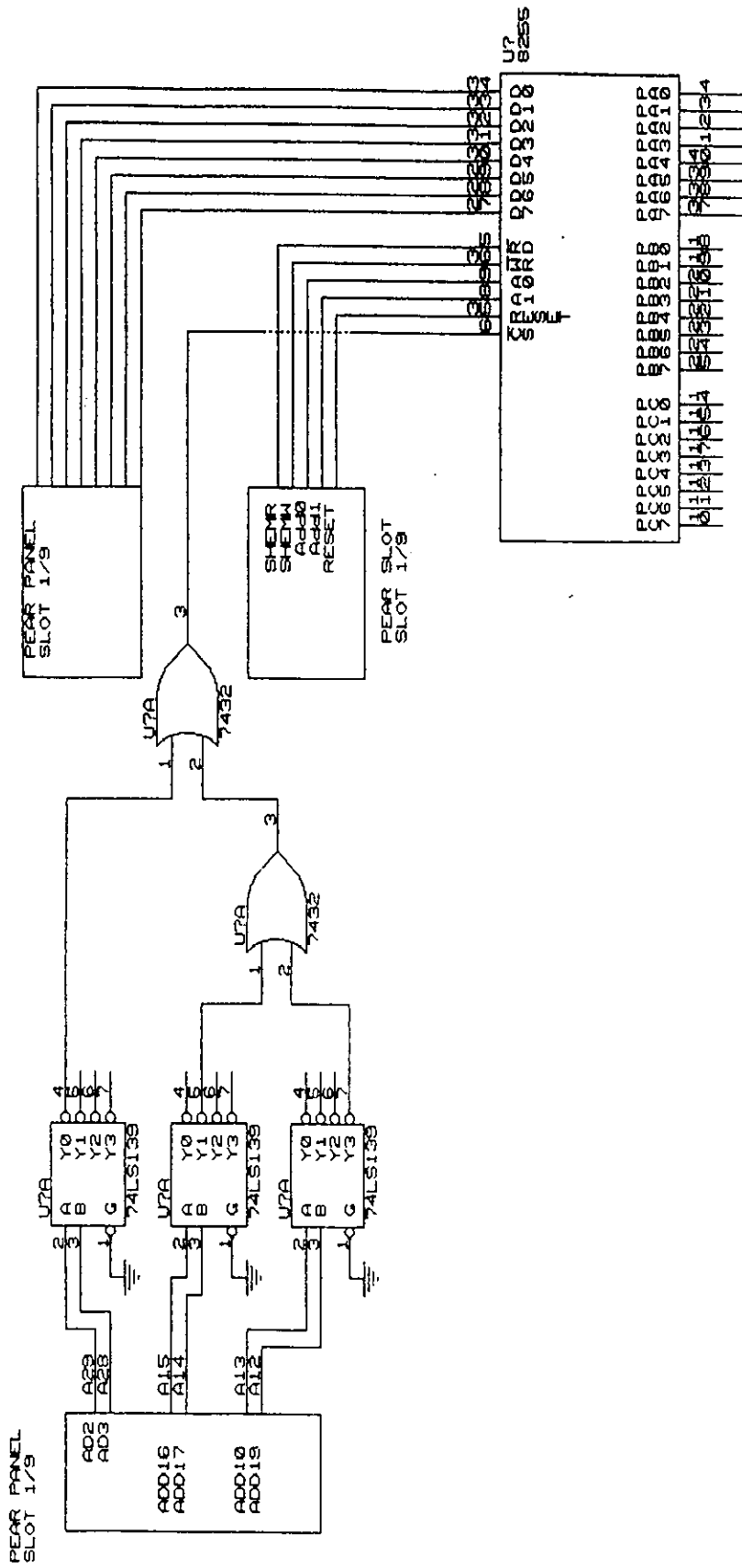


Figura 15

TARJETA DE AISLAMIENTO OPTICO

Para la captura de los dos buses de datos se realizaron dos circuitos equivalentes, el primer circuito conecta el bus de datos de selección de grupo de LED's de U501 del panel de despliegue con el puerto A del 8255. El segundo circuito conecta el bus de datos de 7-segmentos de U501 del panel de despliegue con el puerto C del 8255.

En la figura siguiente se observa cómo está conectada una línea del panel de despliegue con una línea del 8255.

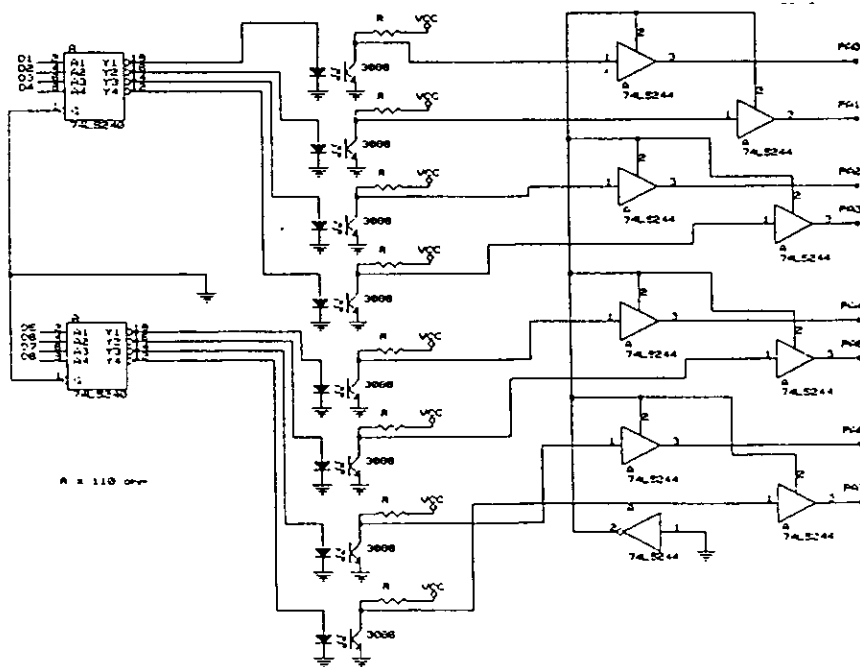


Figura 16

Por ejemplo, el bit D1 del selector de bus del U501 en el panel de despliegue, se conecta a una compuerta inversora 74LS240. La salida del inversor se conecta a un optoaislador 3086. Al tener un cero en la entrada del optoaislador, el diodo del 3086 no conduce. Al no conducir el diodo, el fototransistor entra en región

de corte, y en el colector se tienen los 5 volts de la alimentación. En este momento, ya los circuitos se encuentran ópticamente aislados.

El colector del 3086 se conecta a una compuerta no inversora 74LS244 para luego hacerlo directamente al pin correspondiente en el puerto A (Fig. 17 A).

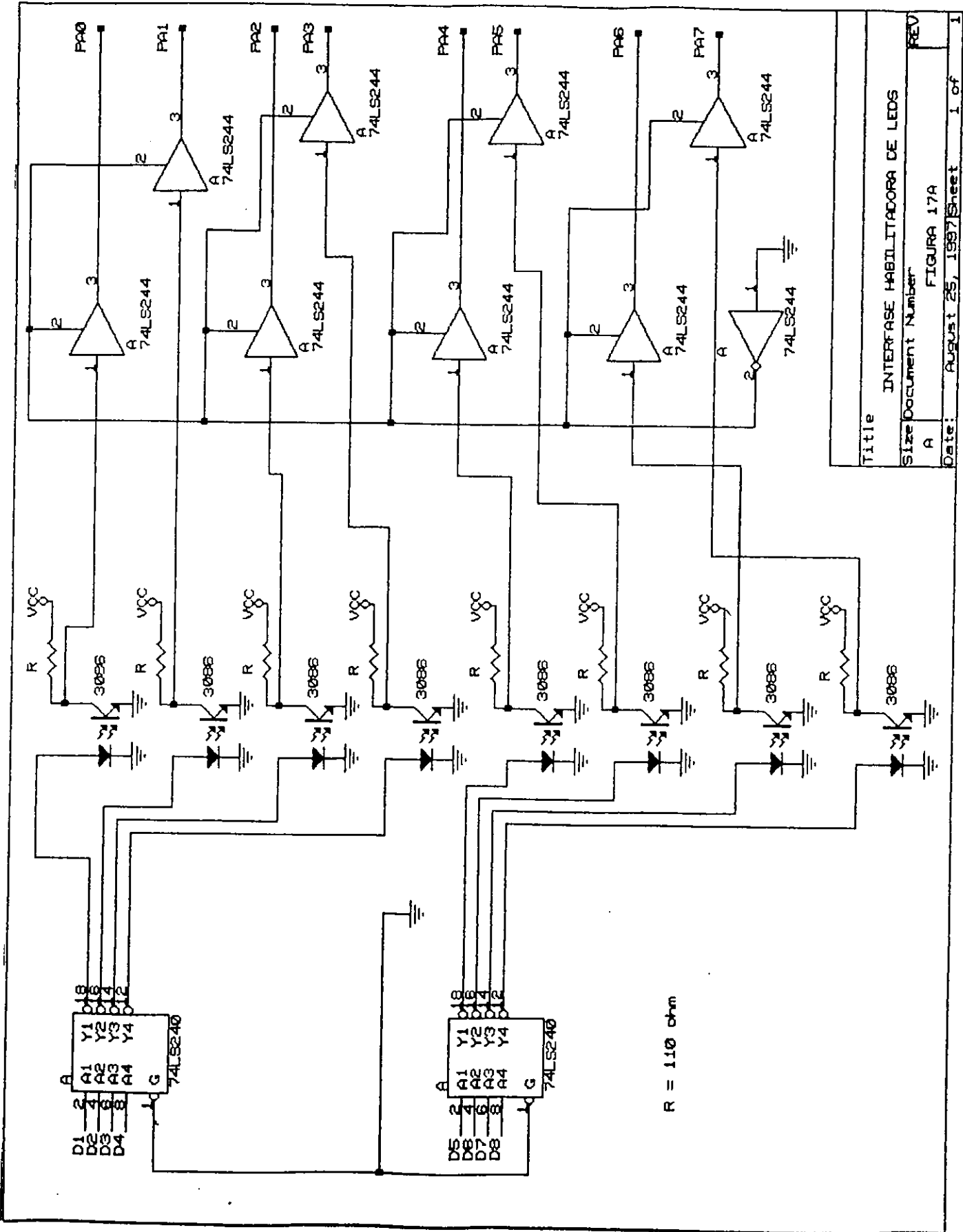
Si a la entrada del optoaislador se tiene un uno, el diodo del 3086 conduce. Si tiene corriente de base en el fototransistor y éste se satura, entonces en el colector hay cero volts. Este circuito es el mismo para los dos buses de datos.

En la Fig. 17B se tiene el diagrama de circuito para conectar el bus de datos de 7-segmentos del U501 con el puerto A del 8255.

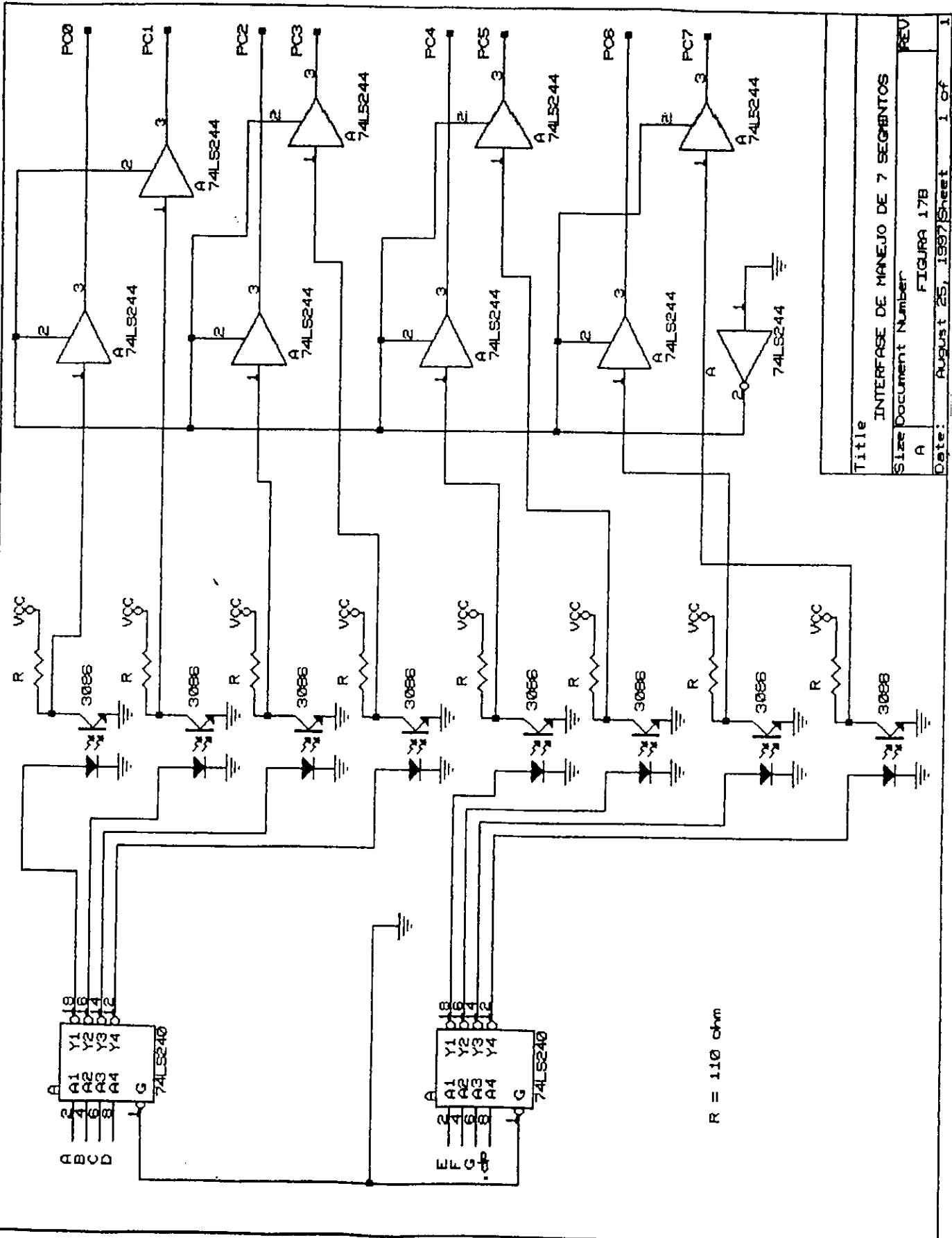
El manejo del teclado del panel de despliegue se hace de la siguiente forma: El puerto B del 8255 manda un código cualquiera, éste entra a una compuerta inversora 74LS240 donde el código es negado. El código ya negado ingresa a la compuerta no inversora 74LS244, por medio de un acoplamiento óptico a través del 3086. Los tres bits menos significativos se conectan a un codificador 3 x 8 74LS138, en el cual cada una de sus 8 salidas ingresa a una compuerta no inversora 74LS244. Las salidas del 74LS244 se van directamente a los botones del panel.

Los bits D4 y D5 que salen del 3086 entran a un conjunto de compuertas AND, cuya función es seleccionar al 74LS244 que está conectado con el teclado del Panel.

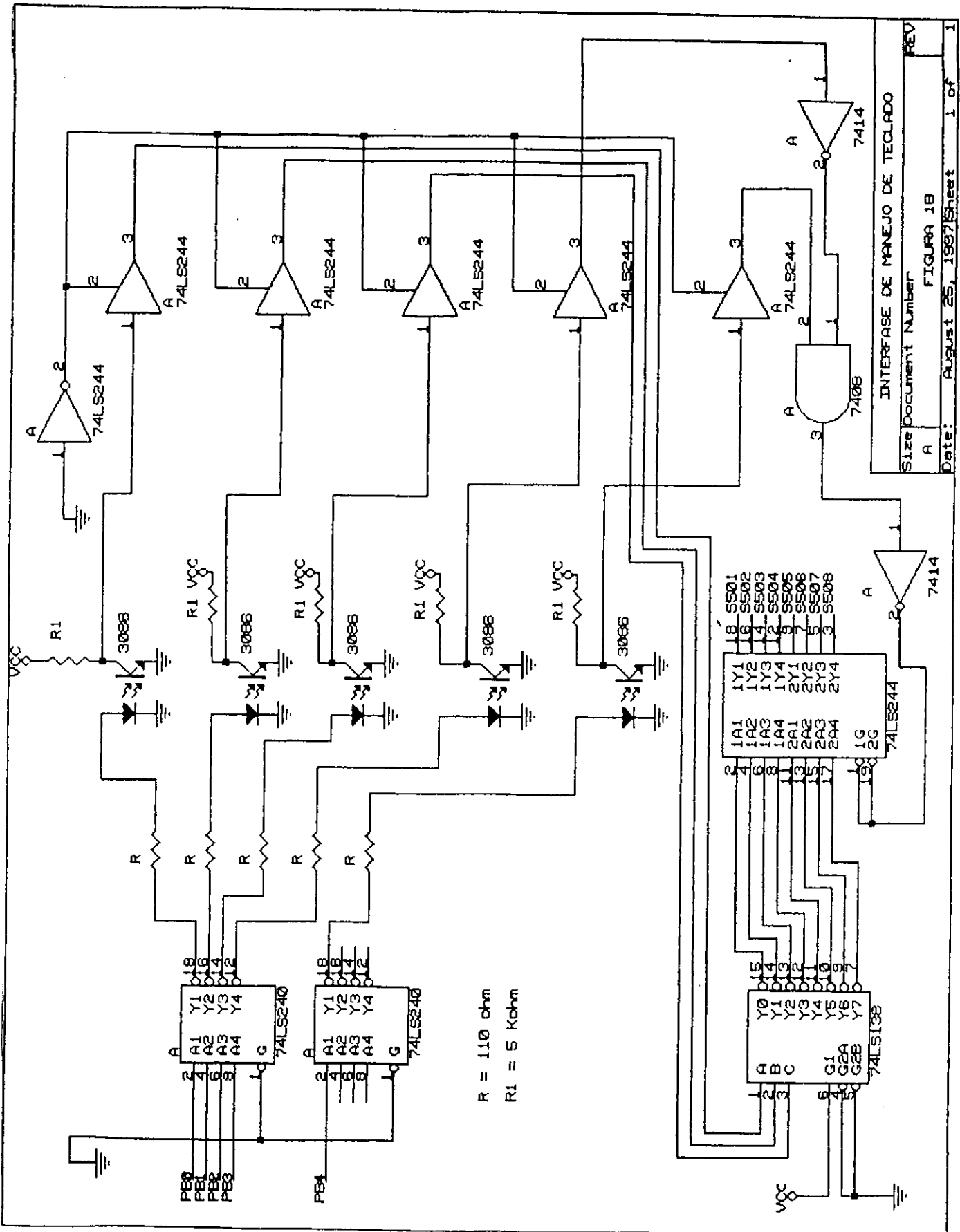
El objetivo es que la salida del 74LS244 esté siempre en alta impedancia cuando se esté usando el teclado del panel, y sólo seleccionar el 74LS244 cuando se necesite pedir información desde la PC. (Fig. 18).



Title	INTERFASE HABILITADORA DE LEDS
Size	A
Document Number	FIGURA 17A
REV	
Date:	August 25, 1987
Sheet	1 of 1



Title		INTERFASE DE MANEJO DE 7 SEGMENTOS
Size Document Number		A
Date:		August 25, 1997
Sheet		1 of 1
REV		FIGURA 17B



INTERFASE DE MANEJO DE TECLADO
 Size Document Number A
 FIGURA 18
 Date: August 25, 1997 Sheet 1 of 1

V. SOFTWARE

Se realizó un programa en Pascal para el manejo de las dos tarjetas. El objetivo es que el programa interprete los códigos con los que el panel de despliegue y la tarjeta de control del UPS se comunican, y así poder capturarlos para poder desplegarlos en la PC. El listado del programa se encuentra en el apéndice.

El programa cumple las siguientes funciones:

- Listado de todos los parámetros del UPS, como voltajes, frecuencias y corrientes.
- Monitoreo constante el UPS para mostrarnos su estado actual.
- Almacenamiento de todos los eventos que ocurran en el UPS, los cuales son salvados en un archivo llamado HISTORY.LST.

La explicación de todas las rutinas que se utilizaron en la elaboración del programa es la siguiente:

- Procedure Status: Se manda a imprimir en pantalla la alarma o el estado del UPS en un determinado momento.
- Procedure Tiempo: Se obtiene la fecha y la hora en la cual ha ocurrido una alarma activa.
- Procedure Print Tiempo: Se imprime en pantalla la fecha y la hora capturadas en el Procedure Tiempo.
- Procedure Archivo: La alarma se guarda en siete arreglos de 200 posiciones , hora, minuto, segundo, año, mes y día en el que ocurrió el evento.
- Procedure Out 8255 B: Se mandan los datos que sean necesarios por el puerto B del 8255, que se encuentra conectado y ópticamente acoplado al teclado del panel de despliegue del UPS.

- Procedure Decodi – Teclado: Se maneja el teclado del panel de despliegue a partir de este procedimiento. Este procedimiento elimina la alta impedancia del buffer 74LS244 que maneja el teclado.
- Procedure IN-8255-A: Se captura los datos que se encuentren en el Puerto A del 8255 en un determinado instante. Este Puerto se encuentra acoplado ópticamente al bus de datos habilitadores de LED's del panel de despliegue.
- Procedure IN-8255-C: Se captura los datos que se encuentren en el Puerto C del 8255 en un determinado instante. Este puerto se encuentra acoplado ópticamente al bus de datos de 7 segmentos del panel de despliegue.
- Procedure Rect-Lect: El dato obtenido por el procedimiento In-8255-C es comparado con una serie de valores que se encuentran en el procedimiento Rect-Lect. Si la comparación no es válida, se siguen tomando datos hasta que sea válida.
- Procedure Conversion: Al dato ya obtenido en el procedimiento anterior se le otorga un valor decimal de 0 a 9, para luego ser desplegado en la pantalla (ver tablas 1 y 2)
- Procedure Pre-Data: Si los datos a desplegar son numéricos, este procedimiento los clasifica en unidades, decenas, centenas o si existe algún punto decimal.
- Procedure Monitoreo: Revisa el Puerto A y el Puerto C para analizar si los datos recibidos son datos numéricos o si representan alguna alarma activa en el UPS.
- Procedure List-Alarm: Guarda en siete arreglos distintos todas las posibles alarmas con sus respectivas fechas, para luego archivarlas y revisarlas.

VI. INSTALACION HARDWARE Y SOFTWARE

La tarjeta optoaisladora se instala adentro el UPS, conectada por medio de un cable plano hacia el panel de despliegue. La tarjeta optoaisladora necesita de 5 volts de alimentación, los cuales se obtienen de la fuente de poder del UPS.

La tarjeta manejadora de puertos del 8255, van hacia algún slot de 8 bits disponible de la computadora, tomando los 5 volts de alimentación de la PC. La tarjeta optoaisladora y la manejadora se conectan por medio de un cable plano, que lleva la información a los 3 puertos del 8255. La tierra de la fuente de poder del UPS y la tierra de la fuente de poder de la PC, no se unen ya que produce ruido en la tarjeta de control del UPS. Por esta razón ambas tarjetas se acoplan ópticamente.

El programa para interpretar los datos del panel de despliegue y manejar el 8255 se realizó en Lenguaje Pascal, con rutinas de Assembler para controlar el 8255.

El programa presenta un menú principal con cinco opciones, las cuales son:

1. Mediciones: Presenta todos los parámetros del UPS los cuales son voltaje de entrada, voltaje de salida, voltaje de baterías, amperios de entrada, amperios de salida, amperios de baterías, frecuencia de entrada y frecuencia de salida.
2. Monitoreo de Alarmas Activas: Nos presenta el estado actual del UPS, indicando si el UPS se encuentra en modo normal o si hay alguna alarma activa. También graba en el archivo HISTORY.LST la fecha y hora de todos los eventos ocurridos.
3. Listado de Alarmas: Presenta en pantalla el historial de las alarmas ocurridas en un determinado período de tiempo. El archivo HISTORY.LST es leído y desplegado en pantalla.

4. Borrado de Alarmas: Borra toda la información existente en el archivo HISTORY.LST.
5. Salida: Se cancela el programa.

Al seleccionar cualquier opción de 1 a 4, se realiza la función requerida y luego al finalizar esta última, se oprime el botón de ENTER y se regresa al menú principal.

VII. CONCLUSIONES

Para interconectar la lógica digital o las señales analógicas con componentes de potencia, se tendrá que realizar por medio de optoaisladores.

El fácil monitoreo de los parámetros y alarmas del UPS se debió a la gran flexibilidad de los modos de operación del 8255.

La velocidad del Z80 del UPS (2mhz) es relativamente baja comparado con la de cualquier PC, por lo que se necesitó colocar retrasos en algunos puntos del programa, para que al UPS procese todas las señales que le mandó la PC.

Al emplear componentes de bajo costo y amplia disponibilidad se puede modernizar un equipo de UPS relativamente antiguo, permitiéndole conectar una PC para el monitoreo de su estado.

VIII. BIBLIOGRAFIA

- EXIDE ELECTRONICS. Schematic Manual Repair for S1000. North Caroline, USA. 1995. 200 pp. Tercera edición.
- EXIDE ELECTRONICS. Descriptions Boards For S1000. North Caroline, USA. 1985 225 pp. Segunda edición.
- EXIDE ELECTRONICS. Schematic Manual Repair For S1200. North Caroline, USA. 1985. 226 pp. Tercera edición.
- EXIDE ELECTRONICS. Description Boards For S1200. North Caroline, USA. 1985. 210 pp. Segunda edición.
- EXIDE ELECTRONICS. Description Boards For S1300. North Caroline, USA. 1986. 300 pp. Segunda edición.
- EXIDE ELECTRONICS. Schematic Manual Repair For S1300. North Caroline, USA. 1986. 324 pp. Primera edición.
- HAMACHER, S. Programmable Peripheral Interface. USA. Mc Graw Hill. 1991. 150 pp. Tercera edición.
- TEXAS INSTRUMENTS INCORPORATED. Transistores Circuitos-Diseños. USA. CECSA. 1990. 278 pp. Tercera edición.
- VRANISISC, M. Organización de Computadoras. USA Prentica Hill. 1992. 125 pp. Cuarta edición.

APENDICE

```
PROGRAM DIGITALS _8255 (INPUT, OUTPUT);
USES CRT,DOS;
TYPE
    ALARMA = RECORD
        TIPO: INTEGER;
        HO,MI,SE,YE,MO,DA,HUND,DWL: WORD;
    END;
    ALARM = ARRAY[ 1..200 ] OF ALARMA;

VAR PAA,PCA,PA,PB,PC,t,X,XI: INTEGER;
    BUENOD,BUENOP,DATA,DNEW,DNEW1,DNEW2,DNEW3:
REAL;
    DATA1,DATA2,DATA3,BUENO,C,D,U: REAL;
    H,OP,J,NI,STA1: INTEGER;
    TIPO1,NA : ARRAY [1..200] OF INTEGER;
    HA,MIA,SEA,YEA,MOA,DAA: ARRAY [ 1..200 ] OF WORD;
    STA,I,SSTA1 : INTEGER;
    HO,MI,SE,YE,MO,DA,HUND,DWL: WORD;
    OPCION,NUM_ALARM: INTEGER;
    ALARMAS: ALARM;
    ARCHIVO: FILE OF ALARM;
    Y: CHAR;

PROCEDURE ABRIR_ARCHIVO;
BEGIN

    FOR NUM_ALARMAS: = 1 TO 200 DO
        ALARMAS[NUM_ALARMAS]. TIPO: = 0;
        {$I-}
        ASSIGN (ARCHIVO, "HISTORY,LST");
        RESET (ARCHIVO) ;
        {$I-}
        IF (IORESULT) <> 0)
            THEN BEGIN
                REWRITE (ARCHIVO) ;
                WRITE (ARCHIVO, ALARMAS);
                NUM_ALARMAS: = 1;
```

```

                END
ELSE BEGIN

    READ (ARCHIVO, ALARMAS) ;
    I: = 0;
    NUM_ALARMA ; = 0;
    REPEAT
        NUM_ALARMA := NUM_ALARMA + 1;
        IF (ALARMAS[NUM_ALARMA].TIPO = 0)
            THEN I: = -23;
        UNTIL ((I = -23) OR (NUM_ALARMA = 200));
        I: = 0;
        END;
END;

PROCEDURE STACK_ALARM;
BEGIN
    FOR I: = 2 TO 200 DO
        ALARMAS[I-1]: = ALARMAS[ I ];
    NUM_ALARMA : = 200;
END;

PROCEDURE CLEAR_ALARM_QUEUE;
BEGIN
    FOR I: = 1 TO 200 DO
        ALARMAS [ I ]. TIPO : = 0;
    NUM_ALARMA : = 1;
END;

PROCEDURE UPDATE_ALARMS;
BEGIN
    IF ((STA > 1) AND (STA < 13))
        THEN BEGIN
            IF NUM_ALARMA = 201 THEN
                STACK_ALARM;
                ALARMAS[NUM_ALARMA].TIPO : = STA;
                ALARMAS[NUM_ALARMA].HO : = HO;

```

```
        IF DATA=120      THEN DNEW:=1;
        IF DATA=0        THEN DNEW:=1;
        IF DATA=16       THEN DNEW:=1;
        IF DATA=64       THEN DNEW:=1;
        IF DATA=255      THEN DNEW:=1;
END;
```

```
PROCEDURE INGRE_COM;
BEGIN
    IN_8255_C;
    DATA:=PC;
END;
```

```
PROCEDURE CIERRE;
BEGIN
    IF BUENOD=1      THEN BUENO:=1;
    IF BUENOP=1     THEN BUENO:=1;
END;
```

```
PROCEDURE INIALIZAR;

BEGIN
    BUENO:=0;
    BUENOD:=0;
    BUENOP:=0;
END;
```

```
PROCEDURE PTO_DEC;
BEGIN
    IF BUENOD=1      THEN C=1;      (apareció decimal)
    IF BUENOP=1     THEN D=1;      (apareció 255)
END;
```

```
PROCEDURE PRE_DATA
BEGIN
    IF D=1          THEN
```

```

        BEGIN
            IF C=1 THEN
                BEGIN
                    DNEW3:= DATA2 + DATA3/10;
                END;
            END;
            IF C=1 THEN
                BEGIN
                    IF D <> 1 THEN
                        BEGIN
                            DNEW3: + DATA1*100 + DATA2 + DATA3/10;
                        END;
                    END;
                END;
            END;
            IF C <> THEN
                BEGIN
                    IF D <> 1 THEN
                        BEGIN
                            DNEW3: + DATA1*100 + DATA2 + DATA3;
                        END;
                    END;
                END;
            END;

```

```

PROCEDURE INIT;
BEGIN
    C:=0;
    D:=0;
END;
PROCEDURE MONITOR;
BEGIN
    IN_8255_A;
    PAA:=PA;
    IN_8255_C;
    PCA:=PC;
END;

```

```

PROCEDURE CONVER_MONITOR;
BEGIN
    TIEMPO;

```

```
UPDATE_ALARMS;  
STATUS;  
DELAY (100);  
END;
```

```
PROCEDURE LIST_ALARM;  
BEGIN
```

```
    CLRSCR;  
    FOR I:=1 TO NUM_ALARMA-1 DO  
        BEGIN  
            IF (SSTA1<> ALARMAS [ I ].TIPO) THEN  
                BEGIN  
                    HO:=ALARMAS[ I ].HO;  
                    MI:=ALARMAS[ I ] MI;  
                    SE:=ALARMAS[ I ] SE;  
                    YE:= ALARMAS[ I ]YE;  
                    MO:=ALARMAS[ I ]MO;  
                    DA:=ALARMAS[ I ]DA;  
                    IF ALARMAS[ I ].TIPO=1 THEN  
                        BEGIN  
                            PRINT_TIEMPO;  
                            WRITELN ( " SYSTEM NORMAL " );  
                        END;  
                    IF ALARMAS[ I ].TIPO=2 THEN  
                        BEGIN  
                            PRINT_TIEMPO;  
                            WRITELN ( " INVERTE FAILURE " );  
                        END;  
                    IF ALARMAS [ I ].TIPO=3 THEN  
                        BEGIN  
                            PRINT_TIEMPO;  
                            WRITELN ( " BATTERY DISCHARGE " );  
                        END;  
                    IF ALARMAS[ I ].TIPO=4 THEN  
                        BEGIN  
                            PRINT_TIEMPO;  
                            WRITELN ( " ON BYPASS " );  
                        END;  
                    IF ALARMAS[ I ].TIPO=5 THEN  
                        BEGIN
```

```
ALARMAS[NUM_ALARMA].MI := MI;  
ALARMAS[NUM_ALARMA].SE := SE;  
ALARMAS[NUM_ALARMA].YE := YE;  
ALARMAS[NUM_ALARMA].MO := MO;  
ALARMAS[NUM_ALARMA].DA := DA;  
NUM_ALARMA := NUM_ALARMA + 1;  
END;
```

```
END;
```

```
PROCEDURE PRINT TIEMPO;  
BEGIN
```

```
    WRITE (YE );  
    WRITE ('/');  
    WRITE (MO);  
    WRITE ('/');  
    WRITE (DA);  
    WRITE (" ");  
    WRITE (HO);  
    WRITE (':');  
    WRITE (MI);  
    WRITE (':');  
    WRITE (SE);  
    WRITE (' ');  
    WRITE (YE);
```

```
END;
```

```
PROCEDURE STATUS;  
BEGIN
```

```
    IF STA = 1      THEN BEGIN  
                    GOTOXY (1,1)  
                    PRINT_TIEMPO;  
                    WITELN ("SYSTEM NORMAL");  
                    END  
    ELSE BEGIN  
                    GOTOXY(1,1)  
                    PRINT_TIEMPO;  
                    WRITELN (' ');  
                    END  
    IF STA = 2      THEN BEGIN  
                    GOTOXY (1,2);
```

```

        PRINT_TIEMPO;
        WRITELN ("INVERTER FAILURE");
        END
ELSE BEGIN
    GOTOXY (1,2);
    WRITELN ( '
END;

IF STA = 3      THEN BEGIN
                GOTOXY (1,3)
                PRINT_TIEMPO;
                WRITELN("BATTERY DISCHARGE");
                END
ELSE BEGIN
    GOTONXY (1,3);
    WRITELN ( '
END;

IF STA = 4      THEN BEGIN
                GOTONXY (1,4)
                PRINT_TIEMPO;
                WRITELN ("ON BYPASS");
                END
ELSE BEGIN
    GOTOXY (1,4);
    WRITELN ( '
END;

IF STA = 5      THEN BEGIN
                GOTOXY (1,5);
                PRINT_TIEMPO;
                WRITELN ("INVERTER NOT READY");
                END
ELSE BEGIN
    GOTOXY (1,5);
    WRITELN ( '
END;

IF STA = 6      THEN BEGIN
                GOTOXY (1,6);
                PRINT_TIEMPO;

```

```

        WRITELN ("SYNC LOSS");
        END;
ELSE BEGIN
    GOTO XY(1,6);
    WRITELN('
END;

IF STA = 7      THEN BEGIN
                GOTOXY (1,7);
                PRINT_TIEMPO;
                WRITELN("INPUT OVERLOAD");
                END
ELSE BEGIN
    GOTOXY(1,7)
    WRITELN ('
END;

IF STA = 8      THEN BEGIN
                GOTOXY (1,8)
                PRINT_TIEMPO;
                WRITELN ("BATTERY FAULT")
                END
ELSE BEGIN
    GOTOXY (1,8);
    WRITELN ('
END;

IF STA = 9      THEN BEGIN
                GOTOXY (1,9);
                PRINT_TIEMPO;
                WRITELN (" AC INPUT FAILURE");
                END
ELSE BEGIN
    GOTOXY (1,9);
    WRITELN ('
END;

IF STA = 10     THEN BEGIN
                GOTOXY (1,10);
                PRINT_TIEMPO;
                WRITELN("LOW BATTERY");

```

```

                END
            ELSE BEGIN
                GOTOXY(1,10);
                WRITELN ('
                END;

        IF STA = 11        THEN BEGIN
            GOTOXY (1,11);
            PRINT_TIEMPO;
            WRITELN ("SHUTDOWN IMMINENT");
            END
        ELSE BEGIN
            GOTOXY(1,11);
            WRITELN ('
            END;

        IF STA = 12        THEN BEGIN
            GOTOXY (1,12);
            PRINT_TIEMPO;
            WRITELN("OVERTEMP");
            END
        ELSE BEGIN
            GOTOXY(1,12);
            WRITELN ('
            END;

END;

```

```

PROCEDURE TIEMPO;
BEGIN
    GETTIME (HO,MI,SE,HUND);
    GETDATE (YE,MO,DA,DWL);
END;

```

```

PROCEDURE ARCHIVOS;
BEGIN
    NA[ I ] = STA;
    HA[ I ] = HO;
    MIA[ I ] = MI;

```

```

SEA[ I ] = SE;
YEA[ I ] = YE;
MOA[ I ] = MO;
DAA[ I ] = DA;
I: = I + 1;
NI: = I;
END;

```

```

PROCEDURE POT_8255_B;
BEGIN

```

```

    ASM
    PUSH ES
    MOV AX,ODOOOH
    MOV ES,AX
    MOV BX,EH      (dir control word)
    MOV AX,99H    (palabra de control)
    MOV ES[BX],AL
    MOV BX,O1H
    MOV AX,PB     (data port)
    MOV ES:[BX],AL
    POP ES
    END

```

```

END;

```

```

PROCEDURE TECLADO;
BEGIN

```

```

    CLRSCR;
    WRITELN ('      "MENU"      ');
    WRITELN ('      1. MEDICIONES GENERALES      ');
    WRITELN ('      2. MONITOREO DE ALARMAS      ');
    WRITELN ('      3. LISTADO DE ALARMAS      ');
    WRITELN ('      4. BORRADO DE ALARMAS      ');
    WRITELN ('      5. SALIDA      ');
    READLN   (OP);

```

```

END;

```

```

PROCEDURE RETARDO;
BEGIN

```

```

    DELAY (100);

```

END

PROCEDURE DECODI_TECLADO;
BEGIN

```
    IF OP=1 THEN
        BEGIN
            PB=19;
            OUT_8255_B;
            PB=16;
            OUT_8255_B
            RETARDO;
            END;
    IF OP=2 THEN
        BEGIN
            PB=19;
            OUT_8255_B;
            RETARDO;
            PB=17;
            OUT_8255_B;
            RETARDO;
            END;
    IF OP=3 THEN
        BEGIN
            PB=19;
            OUT_8255_B;
            RETARDO;
            PB=18;
            OUT_8255_B;
            RETARDO;
            END;
    IF OP=4 THEN
        BEGIN
            PB=20;
            OUT_8255_B;
            RETARDO;
            PB=16;
            OUT_8255_B;
            RETARDO;
            END;
    IF OP=5 THEN
```

```
BEGIN
PB=20;
OUT_8255_B;
RETARDO;
PB=17;
OUT_8255_B;
RETARDO;
END;
IF OP=6 THEN
BEGIN
PB=20;
OUT_8255_B;
RETARDO;
PB=18;
OUT_8255_B;
RETARDO;
END;
IF OP=7 THEN
BEGIN
PB=21;
OUT_8255_B;
RETARDO;
PB=16;
OUT_8255_B;
RETARDO;
END;
IF OP=8 THEN
BEGIN
PB=21;
OUT_8255_B;
RETARDO;
PB=17;
OUT_8255_B;
RETARDO;
END;
IF OP=9 THEN
BEGIN
PB=23;
OUT_8255_B;
RETARDO;
END;
```

END;

PROCEDURE IN_8255_A;
BEGIN

```
    ASM
    PUSH ES
    MOV AX,0D000H
    MOV ES,AX
    MOV BX,3H
    MOV AX,99H
    MOV ES:[BX],AL
    MOV BX,0H
    MOV AL,ES[BX]
    MOV PA,AX
    POP ES
    END
```

END;

PROCEDURE IN_8255_C
BEGIN

```
    ASM
    PUSH ES
    MOV AX,0D000H
    MOV ES,AX
    MOV BX,3H
    MOV AX,99H
    MOV ES:[BX],AL
    MOV BX,02H
    MOV AL,ES[BX]
    MOV PC,AX
    POP ES
    END
```

END;

PROCEDURE RECT_LLECT;
BEGIN

```
    IF DATA=249      THEN BUENO:=1;
    IF DATA=164      THEN BUENO:=1;
    IF DATA=176      THEN BUENO:=1;
```

```
IF DATA=153 THEN BUENO:=1;
IF DATA=146 THEN BUENO:=1;
IF DATA=130 THEN BUENO:=1;
IF DATA=248 THEN BUENO:=1;
IF DATA=128 THEN BUENO:=1;
IF DATA=144 THEN BUENO:=1;
IF DATA=192 THEN BUENO:=1;
IF DATA=121 THEN BUENO:=1;
IF DATA=36 THEN BUENO:=1;
IF DATA=48 THEN BUENO:=1;
IF DATA=25 THEN BUENO:=1;
IF DATA=18 THEN BUENO:=1;
IF DATA=2 THEN BUENO:=1;
IF DATA=120 THEN BUENO:=1;
IF DATA=0 THEN BUENO:=1;
IF DATA=16 THEN BUENO:=1;
IF DATA=64 THEN BUENO:=1;
IF DATA=255 THEN BUENO:=1;
END;
```

```
PROCEDURE CONVERSION;
BEGIN
```

```
IF DATA=249 THEN DNEW:=1;
IF DATA=164 THEN DNEW:=1;
IF DATA=176 THEN DNEW:=1;
IF DATA=153 THEN DNEW:=1;
IF DATA=146 THEN DNEW:=1;
IF DATA=130 THEN DNEW:=1;
IF DATA=248 THEN DNEW:=1;
IF DATA=128 THEN DNEW:=1;
IF DATA=144 THEN DNEW:=1;
IF DATA=192 THEN DNEW:=1;
IF DATA=121 THEN DNEW:=1;
IF DATA=36 THEN DNEW:=1;
IF DATA=48 THEN DNEW:=1;
IF DATA=25 THEN DNEW:=1;
IF DATA=18 THEN DNEW:=1;
IF DATA=2 THEN DNEW:=1;
```

```

        PRINT_TIEMPO;
        WRITELN ( " INVERTER NOT READY" );
    END;
    IF ALARMAS[ I ].TIPO=6 THEN
    BEGIN
        PRINT_TIEMPO;
        WRITELN ( " SYNC LOSS" );
    END;
    IF ALARMAS[ I ].TIPO=7 THEN
    BEGIN
        PRINT_TIEMPO;
        WRITELN ( " OUTPUT OVERLOAD" );
    END;
    IF ALARMAS( I ).TIPO=8 THEN
    BEGIN
        PRINT_TIEMPO;
        WRITELN ( "BATTERY FAULT" );
    END;
    IF ALARMAS[ I ].TIPO=9 THEN
    BEGIN
        PRINT_TIEMPO;
        WRITELN ( " AC INPUT FAILURE" );
    END;
    IF ALARMAS[ I ].TIPO=10 THEN
    BEGIN
        PRINT_TIEMPO;
        WRITELN ( "LOW BATTERY" );
    END;
    IF ALARMAS[ I ].TIPO=11 THEN
    BEGIN
        PRINT_TIEMPO;
        WRITELN ( " SHUTDOWN IMMINENT" );
    END;
    IF ALARMAS[ I ].TIPO=12 THEN
    BEGIN
        PRINT_TIEMPO;
        WRITELN ( " OVERTEMP" );
    END;
    SSTA 1:= ALARMAS [ I ].TIPO;
    END;
    IF I = 25 THEN READLN;

```

```
    IF I = 50 THEN READLN;
    IF I = 75 THEN READLN;
    IF I = 100 THEN READLN;
    IF I = 125 THEN READLN;
    IF I = 150 THEN READLN;
    IF I = 175 THEN READLN;
    IF I = 200 THEN READLN;
    END;
END;
```

```
PROCEDURE MONITOREO;
BEGIN
    BUENO:=0;
    XI:=0;
    REPEAT
    MONITOR;
    XI:= XI + 1;
    IF PAA=16 THEN
        BEGIN
            IF PCA=252 THEN
                BEGIN
                    STA:=1;
                    BUENO:=1;
                    CONVER_MONITOR;
                END;
            END;
        END;
    UNTIL (BUENO = 1= OR (XI=2999);

    BUENO:=0;
    XI:=0;
    REPEAT
    MONITOR;
    XI:XI+1;
    IF PAA=32 THEN
        BEGIN
            IF PCA=243 THEN
                BEGIN
                    STA:=2;
                    BUENO:=1;
                    CONVER_MONITOR;
```

```

        END;
        IF PCA=207 THEN
        BEGIN
        STA:=3;
        BUENO:=1;
        CONVER_MONITOR;
        END;
        IF PCA=63 THEN
        BEGIN
        STA:=4;
        BUENO:=1;
        CONVER_MONITOR;
        END;
    END;
UNTIL (BUENO = 1= OR (XI=2999));

BUENO:=0
XI:=0;
REPEAT
MONITOR;
XI:=XI+1;
    IF PAA=16 THEN
        BEGIN
            IF PCA=63 THEN
                BEGIN
                    STA:=5;
                    BUENO:=1;
                    CONVER_MONITOR;
                    END;
                IF PCA=207 THEN
                    BEGIN
                        BUENO:=1;
                        STA:=6;
                        CONVER_MONITOR;
                        END;
            END;
        END;
UNTIL (BUENO = 1) OR (XI=2999);

BUENO:=0;
XI:=0;
REPEAT

```

```
MONITOR;
XI:=XI+1;
  IF PAA=8 THEN
    BEGIN
      IF PCA=248 THEN
        BEGIN
          STA:=5;
          BUENO:=1;
          CONVER_MONITOR;
          END;
        IF PCA=243 THEN
          BEGIN
            BUENO:=1;
            STA:=8;
            CONVER_MONITOR;
            END;
          IF PCA=207 THEN
            BEGIN
              BUENO:=1;
              STA:=9;
              CONVER_MONITOR;
              END;
            IF PCA=63 THEN
              BEGIN
                BUENO:=1;
                STA:=10;
                CONVER_MONITOR;
                END;
          END;
        END;
      UNTIL (BUENO = 1) OR (XI=2999);

      XI:=0;
      BUENO:=0;
      REPEAT
        MONITOR;
        XI:=XI+1;
          IF PAA=128 THEN
            BEGIN
              IF PCA=248 THEN
                BEGIN
                  BUENO:=1;
```

```

        STA:=11;
        CONVERT_MONITOR;
        END;
        IF PCA=243 THEN
        BEGIN
        STA:=12;
        BUENO:=1;
        CONVERT_MONITOR;
        END;
        END;
UNTIL (BUENO=1= OP (XI=2999));
END;

BEGIN
    ABRIR_ARCHIVO;
    I:=1;
    NI:=1;
    REPEAT
        TECLADO;
        CLRSCR;
        IF (OP = 1) THEN
        BEGIN
            OPCION:=0;
            REPEAT
                OPCION:=OPCION + 1;
                IF OPCION = 9
                THEN OPCION:=1;
                DECODI_TECLADO;
                INIT;
                INICIALIZAR;(EMPIEZA LECTURA DE LA CENTENAS)
                J:=0;
                REPEAT
                    J:J+1;
                    INIT
                    INICIALIZAR;
                    DECODI_TECLADO;
                    REPEAT
                        REPEAT
                            IN_8255_A;
                        IF PA=1 THEN

```

```

        BEGIN
        INGRE_COM;
        RECT_LECT;
        CIERRE;
        END;
    UNTIL PA=1;
UNTIL BUENO=1;
PTO_DEC;
CONVERSION; (FINALIZA LECTURA DE LAS CENTENAS)
DATA1:=DNEW;
INICIALIZAR; (EMPIEZA LECTURA DE LAS DECENAS)
REPEAT
    REPEAT
    IN_8255_A;
    IF PA=2 THEN
        BEGIN
        INGRE_COM;
        RECT_LECT;
        CIERRE;
        END;
    UNTIL PA=2;
UNTIL BUENO=1;
PTO_DEC;
CONVERSION; (FINALIZA LECTURA DE LA UNIDADES DECENAS)
DATA2:=DNEW;
INICIALIZAR; (EMPIEZA LECTURA DE LAS UNIDADES)
REPEAT
    REPEAT
    IN_8255_A;
    IF PA=4 THEN
        BEGIN
        INGRE_COM;
        RECT-LECT;
        CIERRE;
        END;
    UNTIL PA=4;
UNTIL BUENO=1;
PTO_DEC;
CONVERSION; (FINALIZA LECTURA DE LAS UNIDADES)
DATA3=DNEW;
PRE_DATA;

```

```

                UNTIL J=4;
IF OPCION =1
THEN BEGIN
    GOTOXY(1,1);
    WRITELN ('VOLTAJE DE ENTRADA:           ',DNEW3:10:2);
    END;
IF OPCION =2
THEN BEGIN
    GOTOXY(1,2);
    WRITELN ('VOLTAJE DE SALIDA:             ',DNEW3:10:2);
    END;
IF OPCION =3
THEN BEGIN
    GOTOXY(1,3);
    WRITELN ('VOLTAJE DE BATERIAS:                 ',DNEW3:10:2);
    END;
IF OPCION =4
THEN BEGIN
    GOTOXY(1,4);
    WRITELN ('AMPERIOS DE ENTRADA:                 ',DNEW3:10:2);
    END;
IF OPCION =6
THEN BEGIN
    GOTOXY(1,5);
    WRITELN ('AMPERIOS DE SALIDA:                 ',DNEW3:10:2);
    END;
IF OPCION =5
THEN BEGIN
    GOTOXY(1,6);
    WRITELN ('AMPERIOS DE BATERIA:                 ',DNEW3:10:2);
    END;
IF OPCION =7
THEN BEGIN
    GOTOXY(1,7);
    WRITELN ('FRECUENCIA DE ENTRADA:              ',DNEW3:10:2);
    END;
IF OPCION =8
THEN BEGIN
    GOTOXY(1,8);
    WRITELN ('FRECUENCIA DE SALIDA:                ',DNEW3:10:2);
    END;

```

```
DELAY (1000);
UNTIL KEYPRESSED;
END;
IF OPCION =2
  BEGIN CLRSCR;
    REPEAT
      MONITOREO;
      UNTIL KEYPRESSED;
      Y:=READKEY;
      CLRSCR;
    END;

  IF OP=3 THEN
    BEGIN
      LIST_ALARM;
      WRITE('PRESIONE UNA TECLA PARA CONTINUAR...');
      READLN;
    END;

  IF OP=4 THEN
    CLEAR_ALARM_QUEUE;
  UNTIL OP=5

  REWRITE (ARCHIVO);
  WRITE (ARCHIVO,ALARMAS);
  CLOSE (ARCHIVO);
  END.
```